A METHOD FOR SYSTEM OF SYSTEMS DEFINITION AND MODELING USING PATTERNS OF COLLECTIVE BEHAVIOR

A Dissertation
Presented to
The Academic Faculty

by

Eric Inclan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2021

A METHOD FOR SYSTEM OF SYSTEMS DEFINITION AND MODELING USING PATTERNS OF COLLECTIVE BEHAVIOR

Approved by:

Dr. Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Michael J. Steffens
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Daniel P. Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Jean Charles Domerçant
Georgia Tech Research Institute

Dr. Santiago Balestrini-Robinson
Georgia Tech Research Institute

Date Approved: April 29, 2021

DEDICATION

To my family

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Dr. Mavris for his sense of humor, patience, leadership, and support. Your feedback has always been as accurate and valuable as it has been incisive. I will undoubtedly be a better scientist and engineer for it, and I am grateful for all the guidance you have given me.

I want to thank all of the Georgia Tech faculty and staff that I had the privilege to learn from and work for, for embodying a level of professionalism, talent, depth of understanding, empathy, and diligence that is very hard to find, let alone in one place.

I would like to thank my lab-mates, and in particular John Mines, Stephanie Zhu, Arturo Santa, and Katie Handschuh Fuerst, for listening when I needed someone to talk to. I would also like to thank Stephanie Zhu for the late nights we spent debating the finer points of fluid mechanics and pouring over this and that equation.

I would like to thank my mother, my wife, and my son for their love and for inspiring me to press forward.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

AoA	Analysis of Alternatives
ABM	Agent-Based Modeling
CBA	Capabilities-Based Assessment
DAS	Defense Acquisition System
DoD	Department of Defense
DOTLmPF-P	Doctrine, Organization, Training, Leadership and Education, Materiel, Personnel, Facilities, and Policy
DSM	Design Structure Matrix
INCOSE	International Council on Systems Engineering
IPS	Integrated Product Support
IRMA	Interactive Reconfigurable Matrix of Alternatives
JCIDS	Joint Capabilities Integration and Development System
JROC	Joint Requirements Oversight Council
M&S	Modeling and Simulation
NAVSEA	Naval Sea Systems Command
RGS	Requirements Generation System
SAR	Search and Rescue
SE	Systems Engineering
SoSE	Systems of Systems Engineering
USN	United States Navy

SUMMARY

The Department of Defense ship and aircraft acquisition process, with its capability-based assessments and fleet synthesis studies, relies heavily on the assumption that a functional decomposition of higher-level system of systems (SoS) capabilities into lower-level system and subsystem behaviors is both possible and practical. However, SoS typically exhibit "non-decomposable" behaviors (also known as emergent behaviors) for which no such representation exists. The presence of unforeseen emergent behaviors, particularly undesirable ones, can make systems vulnerable to attacks, hacks, or other exploitation, or can cause delays in acquisition program schedules and cost overruns in order to mitigate them. The International Council on Systems Engineering has identified the development of methods for predicting and managing non-decomposable behaviors as one of the top research priorities for the Systems Engineering profession. Therefore, this thesis develops a method for rendering non-decomposable, quantifiable SoS properties and behaviors traceable to patterns of interaction of their constitutive systems, so that exploitable patterns identified during the early stages of design can be accounted for. This method is designed to fill two gaps in the literature. First, the lack of an approach for mining data to derive a model (i.e. an equation) of the non-decomposable behavior. Second, the lack of an approach for qualitatively and quantitatively associating non-decomposable behaviors with the components that cause the behavior.

In order to facilitate the development of this method, this research relies on a model-based framework. It adopts an existing model-based definition of the term "system," and then sets out to reconcile portions of the literature on complex behaviors, emergent

behaviors, and systems of systems, in terms that are amenable to models and the modeling process. This research then studies the nature and limitations of modeling and simulation (with an emphasis on agent-based modeling). Within the confines of two carefully qualified assumptions (that the model is valid, and that the model is efficient), it is argued that simulated emergence is bona-fide emergence, and that simulations can be used for experimentation without sacrificing rigor.

Systems that self-organize are determined to have a physical structure that is non-decomposable (this includes physical, chemical, and biological systems, as well as many military conflicts and some man-made systems). The first hypothesis proposed in this thesis is that self-organized structures imply the presence of data compression, and this compression can be used to explicitly calculate an upper bound on the number of non-decomposable behaviors that a system can possess.

Non-decomposable behaviors are referred to as emergent behaviors, while the term complexity is reserved for the measurable quantities associated with predicting a system's capacity for and expression of emergence. Therefore, in this thesis, emergence is not a measurable quantity, but rather, it is the byproduct of the organization / structure of a collection of components, and is only observable when two systems interact. This thesis collects seven necessary conditions and presents a pragmatic definition for emergent behavior. The second hypothesis proposed in this thesis is that a set of numerical criteria for detecting emergent behavior constitute sufficient conditions for identifying weak and functional emergent behaviors.

This thesis then applies the method to a simulated flock of birds, a notional aerial combat model, and simulation of swarms of unmanned quadcopter drones. A third hypothesis is proposed, which states that targeting the system-level properties of these self-organized systems can be more effective than affecting any given component of the system, according to a problem-specific measure of merit. Using the method developed in this thesis, exploitable properties are identified and component behaviors are modified to attempt the exploit.

These tests results find that Hypothesis 2 is falsified, and that the numerical criteria are not sufficient conditions after finding instances that produces a false-positive. As a result, a set of sufficient conditions for emergent behavior identification remains to be found. Therefore, the test for Hypothesis 1 was conducted based on a worst-case scenario where the largest possible number of obtainable emergent behaviors was compared against the limit computed from the smallest possible data compression of a self-organized system. Based on this conservative test, Hypothesis 1 was also falsified. Hypothesis 3, on the other hand, was supported, as it was found that new behavior rules based on component-level properties provided less improvement to performance against an adversary than rules based on system-level properties. Overall, the method is shown to be an effective, systematic approach to non-decomposable behavior exploitation, and an improvement over the modern, largely ad hoc approach.

CHAPTER 1. INTRODUCTION AND BACKGROUND

1.1 Foreword

The scope of this thesis lies at the intersection of studying collective dynamical behaviors of independent systems, systems science, ***Systems of Systems Engineering*** (SoSE), ***Modeling and Simulation*** (M&S),[1] and the study of emergent behavior, all of which are extensively studied fields of fundamental importance to the fleet synthesis studies conducted by the ***United States Navy*** (USN). The study of emergent behavior, however, permeates nearly every field of modern science (depending on the definition one ascribes to). Therefore, the narrative of this document must be structured in a somewhat unorthodox way. Rather than begin with what could be a very abstract, meandering review of the literature associated with emergence, it will first introduce the practical, engineering application of this thesis. That is, it will begin with the most narrow, down-to-earth scope possible. By the end of Chapter 1, the subject matter will have already become more abstract, but still remain within the realm of practical engineering questions. Chapters 2-4 gradually expand this scope starting from a notional, USN-specific case study, then transitioning into an abstract, canonical case study, and concluding with a purely abstract discussion of this author's definition of emergent behavior. Those chapters introduce the hypotheses underlying this thesis. Chapter 5 will then describe the experiments that will be conducted to support or falsify the hypotheses. The remaining chapters characterizing the results will then explain the consequences of the experimental findings, beginning with the canonical case, and concluding with more practical cases.

[1] See also Model-Based Systems Engineering [274] [275]. The term ***model*** will be rigorously defined in CHAPTER 2. For the current chapter the reader can safely assume that it is simply a set of mathematical equations implemented by a computer to predict the possible outcome of some scenario.

1.2 Department of Defense Acquisition

The ***Department of Defense*** (DoD) defines ***acquisition*** as "the conceptualization, initiation, design, development, test, contracting, production, deployment, integrated product support (IPS), modification, and disposal of weapons and other systems, supplies, or services (including construction) to satisfy DoD needs, intended for use in, or in support of, military missions" [1]. As the definition suggests, the DoD acquisition process is a months-to-years long[2] cascading interplay between decision-makers and the analysts that generate knowledge for the decision-maker[3]. Each procurement effort is often performed by distinct groups of individuals within the acquisition community [2], and the final product is part of a strategic portfolio [3] [4] of "means and ways"[4] that satisfy some mission objective(s). Here, ***means*** is defined as "military resources (manpower, materiel[12], money, forces, logistics, and so forth) required to accomplish the mission", while ***ways*** is defined as "the various methods of applying military force… [and therefore] courses of action designed to achieve the military objective" [5][5]. Therefore, although a materiel[12] procurement effort may concern itself with a single weapon or a single ship (a specific means), what the USN is ultimately assembling is a fleet. It will be shown (Sections 1.3-1.5) that a fleet can be treated as a means in and of itself, which creates new ways of achieving objectives distinct from those enabled by its constitutive ships and weapon

---

[2] To provide a sense of duration, consider that simpler, auxiliary ships can be built in less than two years [232] while the more complex ship designs require ten years to design and over five years for construction [229].

[3] The interested reader is referred to Figure 10 of [30] to get a sense of how many parties are (in)directly involved in ship design, bearing in mind that design is just one component of acquisition. For a deeper discussion of the issues facing decision-makers, see the second chapter of [56].

[4] The terms "means and ways" carry special significance in military documentation, as they comprise parts of the definition of the term ***strategy***, which originated in [5]. Interested readers are referred to [237] [238] for further context and criticisms.

[5]Lykke subsequently states "These courses of action are termed 'military strategic concepts'," [5] which bears the unfortunate consequence of rendering Lykke's definition of ***strategy*** at least somewhat circular. Meiser clarifies, "In practice ways are simply the actions to be taken using the resources available to achieve a goal," which he then argues is detrimental to strategic thinking [238].

systems. While this may seem obvious, the study of these new ways is one of the biggest challenges currently facing analysts in the acquisition community (see Sections 1.5-1.7).

In general, the procurement-driven portion[6] of the modern acquisition process is punctuated by three major decision-making ***Milestones***, where Milestone A is the decision to invest in new technology development, Milestone B is the decision to proceed with engineering / manufacturing, and Milestone C is the decision to end prototyping[7] and begin full-scale production [6].[8] The phases of the modern acquisition process[9] (highlighting the ***Defense Acquisition System***, DAS [7], and ***Joint Capabilities Integration and Development System***, JCIDS [8] [9]) are depicted in Figure 1[10] (reproduced from [8]).

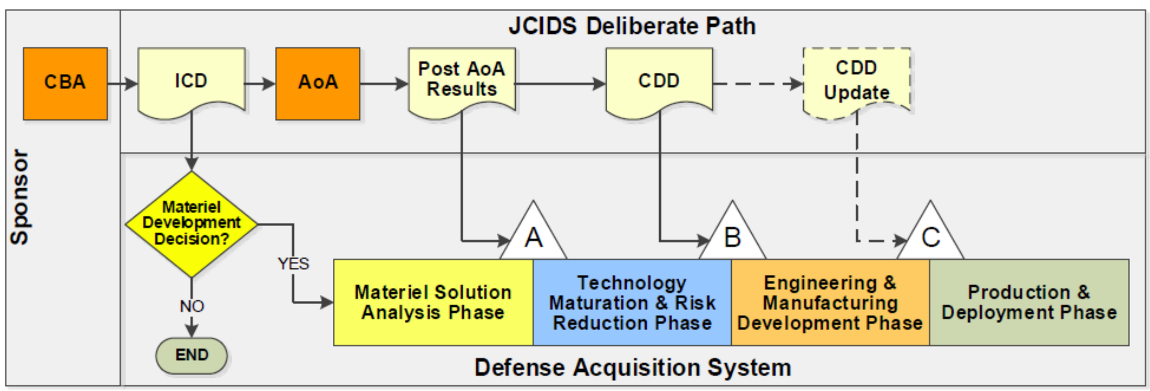


Figure 1 –The Major Phases of the Defense Acquisition Process

---

[6] As opposed to the operations and support phase, which includes the sustainment and ultimate disposal of procured equipment, and begins once full-scale production has been reached (see Figure 3 in [7]).

[7] One noteworthy exception is ship building, which traditionally has no distinct prototyping phase the way aircraft procurement does. That is, each ship prototype is also the final product [6] [231] [244]. Furthermore, each subsequent ship in a class is usually a modification of the previous prototype, meaning very few ships are actual copies [246]. Progressive modifications are sometimes termed ***evolutionary acquisition*** [245].

[8] More detailed information on additional tasks and decision-making concerns in the lead up to Milestones A and B are discussed in [230] [13] [233] [248].

[9] The three major components of modern DoD acquisition are the DAS, the ***Planning, Programming, Budgeting, and Execution Process*** (PPB&E), and the JCIDS [236] [7]. Although each is logistically significant, the JCIDS provides enough exposition to cover the concepts relevant to this thesis.

[10] The acquisition process depicted in Figure 1 can vary. As shown, the ***Initial Capabilities Document*** (ICD) and the ***Capability Development Document*** (CDD) are documents that inform subsequent stages of acquisition. However, some programs allow for waivers to this (or associated) documentation, while in other programs, a single ICD can result in multiple CDDs being generated. Readers interested learning more about canonical deviations from the standard process are referred to [8] [9] [235].

Prior to Milestone A are the ***Capabilities-Based Assessment*** (CBA) and the ***Analysis of Alternatives*** (AoA). Enclosure E of the Charter of the ***Joint Requirements Oversight Council*** (JROC) and the Implementation of the JCIDS defines a ***capability***[11] as "the ability to achieve a desired effect under specified standards and conditions through a combination of means and ways across Doctrine, Organization, Training, Leadership and Education, Material [sic],[12] Personnel, Facilities, and Policy,"[13] [9] while ***materiel***[12] is defined as "equipment, apparatus, and supplies used by an organization or institution" [1]. In short, the purpose of the CBA is to "assess the capabilities" the USN has or needs, "identify any gaps [in capabilities] that might exist and make recommendations for how those gaps can be closed" [10]. If the group conducting the CBA recommends a materiel solution to close a gap, an AoA is conducted.[14] The purpose of the AoA is to "… identify, evaluate, and document the costs and mission effectiveness of alternative [materiel solutions] … help establish critical mission characteristics and … performance requirements, and then identify potential alternative [materiel solutions] that can satisfy those requirements" [11]. Furthermore, the AoA "considers the sensitivity of each alternative to possible changes to key assumptions or variables" [12]. Although it is anachronistic to apply the current procurement process to twentieth century USN acquisition, the acquisition community has always performed similar analyses in one form

[11] This definition is close to the Oxford dictionary definition, "the power or ability to do something," [239] which can also be used for the purposes of this thesis without loss of generality. Various authors of military literature use the term capability differently. Example 1: The phrase "amount of capability" appears to be common jargon [241] [242]. Example 2: Similar to Oxford include [243] [8] [1], and the glossary of [9].
[12] The term "Material" in the definition of capability on page E-1 of the charter should be spelled "Materiel" as it is in other places [9]. Material generally refers to raw resources, while ***materiel*** refers to the completed equipment the acquisition community procures [1].
[13] ***Doctrine, Organization, Training, Leadership and Education, Materiel, Personnel, Facilities, and Policy*** (DOTLmPF-P [9]; formerly DOTLMPF [240]) is essentially the list of options for ways and means that the acquisition community considers when deciding how to best achieve a particular goal. See Section 3.11.5 of [36] for a discussion of the acronyms, and additional clarification.
[14] See [247] for examples of challenges faced in the transition from CBA to AoA.

or another. What has changed substantially, however, is the guiding philosophy of USN acquisition.[15]

1.3 Paradigm Shifts in USN Ship Design, Acquisition, and Modeling

During much of the twentieth century,[16] DoD acquisition community set its requirements using a heavily threat-based approach, referred to as the Requirements Generation System (RGS) [13]. Relative to modern acquisition, this approach was reactive, being driven in part by enemy military capabilities[17] [14]. The advantage of a reactive approach is that the perceived capability gap is restricted to a specific, unmet threat, enabling designs to be tailored for effectiveness (also reducing the likelihood of requirements creep).[18] On the other hand, limited scopes and performance-driven design techniques often resulted in the creation of overly specialized designs that were very expensive but seldom used [15] [16] [17]. Thus, one of the major paradigm shifts in USN acquisitions was a pivot toward development of less specialized, multi-mission / multi-role ships, including a push for so-called ***modular*** ships[19] that can support interchangeable systems [18] [19] [20] [21].

An important consequence of the modularity paradigm shift was the increased focus on system[20] interoperability [22] [23]. The DoD defines a ***system*** very generally as "a

---

[15] A useful observation, provided here for completeness, is that "[the] fundamental shift toward capability-based acquisition and design is best described by a shift away from 'things' to 'ways to do things'," [273].

[16] Readers interested in more detailed historical information about the DoD acquisition process are referred to the reviews provided in [56] [59], as well as the e-books [23] [227], and the paper [14].

[17] For a nuanced discussion of the distinction between the two frameworks (particularly, the distinction between designing for required capabilities rather than hitting a specific system[20] requirement) see Section 1.2.1 of [56].

[18] Requirements creep is defined as "the tendency of the user (or developer) to add to the original mission responsibilities and/or performance requirements for a system while it is still in development," [1] and is a frequently-cited cause of acquisition cost overruns [248] [6] [2]. Note that in a 2002 memo reproduced in [10], then Secretary of Defense Donald Rumsfeld criticized the RGS for "[requiring] things that ought not to be required," which highlights a misalignment of requirements, but not necessarily requirements creep.

[19] The notion of modularity has recently been superseded by "adaptability." See [16] [249] [250].

[20] For this section, system refers to the major components and equipment onboard a ship such a weapon, or the hull, as in [28] [22]. A *subsystem* usually refers to minor components that independently perform a very specific function [1].

functionally, physically, and/or behaviorally related group of regularly interacting or interdependent elements; that group of elements forming a unified whole," [24] and ***interoperability*** as "1. The ability to act together coherently, effectively, and efficiently to achieve tactical, operational, and strategic objectives" [24]. Although early publications regarding interoperability focus on communication [22] and software [25], the study of interoperability was ultimately generalized to accommodate any interaction between two systems [25] [26] [27]. Thus, it was recognized that the ability of systems (weapon systems, hull structures [28], etc.) to interact effectively could impact overall ship performance just as much as the performance of the individual systems in isolation [29]. Furthermore, multi-mission ships must compete effectively in different operating environments (or, at least, against a broader set of adversaries), which increases the range of potential interactions systems can be exposed to. The topic of system interactions will be elaborated on in Sections 1.5-1.7, and Section 2.3.

Another major paradigm shift (one that accompanied technological advances in computing) was the acquisition community's increasing use of modeling and simulation (M&S) to inform each stage of acquisition,[21] even going so far as to model the design [30] and acquisition processes themselves [23] [31]. Particularly within the design stages of acquisition, entire computational environments have been created where models of smaller components feed their results into models of ensembles, which feed into models of ensembles of ensembles.[22] While this has substantially cut costs [23], these models are often developed independently, which may result in sets of models that have incompatible underlying assumptions and varying accuracy, thereby producing illogical/invalid results.[23]

---

[21] Examples of models developed with applications in defense acquisition include: data gathering [156], requirements setting [151] [254], optimization [252], forecasting [258], and decision-making [150].
[22] For example: AFSIM [251]; ASSET, LEAPS, and IHDE among others [253]; see also for the aforementioned and discussion of S3D [255] [256].
[23] See [257] for frameworks that verify the composition of model ensembles. See [259] [260] [261] for studies of combined models with varying accuracy. See [262] for pitfalls in simplifying one popular class of models.

If subsystem[20] behaviors are incorrectly modeled, then quantitative and qualitative predictions of their interactions will suffer (or fail entirely), which diminishes the analyst's ability to predict the overall behavior of the system (see Sections 1.5-1.7). These issues undermine both the ability to identify a capability gap (via M&S), as well as the ability to identify an appropriate set of materiel solutions.[24]

As suggested earlier, the most recent and significant paradigm shift within the acquisition community was the transition to the capabilities-based, proactive approach [10] briefly introduced in Section 1.2. Chronologically, the implementation of JCIDS occurred decades after the world had effectively become militarily unipolar.[25] Thus, a consequence of the shift, is that "proactive" took on the additional meaning of developing and maintaining military overmatch (i.e. superiority) [32] [33] [34] [35]. Doing so requires a rapid pace of adaptation,[26] reliable forecasting of the enemy's evolving capabilities (typically performed over 30-year windows), and the ability to consider multiple solutions of various types. One purpose of the CBA is precisely to examine candidate DOTLmPF-P[13] solutions.

As a somewhat simplistic example, consider that a nation-state threat to national security creates a gap that can be filled through diplomatic negotiations with the potential adversary, or military deterrence (as is achieved via overmatch). Within the scope of diplomacy, relationships with other nations can be improved by strengthening economic ties, or via gestures of goodwill such as providing disaster relief. Search and rescue missions commonly occur during disaster relief. A destroyer capable of launching helicopters can aid in search and rescue missions as well as the power projection[27] required

[24] Even existing materiel solutions can require modeling during the CBA and AoA [56].
[25] By some estimates, the USN currently dominates its closest competitor by a large margin [263] [264] [265], although concerns of future "near-peer" conflict have been voiced [267] [266] [268]. For timeline information, see [23] [56] [81].
[26] See introduction section of [269]
[27] See [32] for use of the phrase "power projection"

for military deterrence. Thus, the destroyer and helicopters considered for acquisition can play a direct role in deterrence, or a supporting role in diplomatic missions, meaning that the performance of these vehicles must be taken into account when evaluating capabilities for either mission. As illustrated in this example, a second consequence of this shift is that the acquisition community is now encouraged to consider more information as a matter of practice. For example, the Air Force CBA Handbook states, “When developing solutions, the study team should consider system-of-systems[28] and family-of-systems.[28] One common error is fixating on one aspect of a system. Many times, complex[28] problems are best solved by making moderate improvements to multiple systems rather than a single, large improvement to one system” [36].

Both of the aforementioned consequences compound the need to fill a cascade of knowledge gaps. In a threat-based approach, the destroyer might only have to be modeled for two missions, which could possibly extend to include modeling the strike group the destroyer embarks on missions with. However, in a capabilities-based approach, this destroyer would have to be modeled in those two contexts, as those contexts evolve over the next 30 years. This includes possible new technologies and upgrades for the destroyer, technology/ship replacements for the strike groups, and changes to the operating environment. Thus, each acquisition now requires knowledge of the technologies / systems being developed (present and future), as well as the ship designs the systems will be placed into, as well as the fleets (present and future) that the ship operates in, in order to make an acquisition decision.[29] As a result, the USN gradually expanded the scope of the analysis used in JCIDS.

---

[28] The term ***system-of-systems*** will be formally introduced in Section 1.5. For the current narrative, it suffices to think of it as a collection of systems (e.g. a fleet of ships). The various definitions of the term ***complex*** will be discussed in Section 1.5 and CHAPTER 4. The term ***family of systems*** is defined in [8] [46], briefly discussed in [271], and is outside the scope of this thesis.

[29] A variety of concerns have been raised regarding this new acquisition process. For criticisms/limitations of the current approach, including calls to overhaul it, see [226] [228] [267] [247] [235] [234].

1.4 Navy Fleet Synthesis Studies

The CBA was first introduced with JCIDS in 2002. By 2010 there were calls for the CBA to accommodate fleet synthesis studies [37]. A fleet synthesis[30] study, for the purposes of this thesis, will be defined as any study that "involves constructing alternative views of the future, then setting up and tracking the resulting course of [fleet composition] evolution from the present fleet to a long-run future state." [38]. Although fleet synthesis studies were not new [39] [40], the first study of long-term fleet synthesis performed by ship designers at the Naval Sea Systems Command (NAVSEA) began in 2006 [41].[31] Unlike previous studies, the objective of this study "was not just to design ships, but to design the entire Navy and to program a ship-by-ship and year-by-year transition from the currently planned fleet to an alternative fleet… This integration of surface ship fleet mixes, ship designs, and long-range evolutionary planning, is new." [41] This is the systems of systems perspective later adopted by the Air Force CBA Handbook (Section 1.3). Every fleet synthesis study requires modeling systems of systems (Section 1.5).

As previously stated, system interactions (i.e. interoperability) play a critical role in ship and, ultimately, fleet performance. Some of the aforementioned studies incorporate quantifiable parameters that augment/diminish fleet performance as a mathematical representation of the impact of coordinated ship interactions (e.g. "presence multipliers" [38]). However, fleet synthesis studies explicitly incorporating quantifiable measures of interoperability do not seem to appear in the literature for at least another ten years after the NAVSEA study [29]. This suggests that many fleet synthesis studies have not

[30] See [29] [38] for discussions of related terms including: fleet mix, force structure, and naval architecture.
[31] The report of the original "Affordable Future Fleet Study" does not appear to be publicly available.

accounted for ship-interaction-dependent capabilities in a readily traceable way, which can result in subsequent fleet-level capabilities being excluded or misrepresented. At least one fleet synthesis study altogether assumed “capabilities of a fleet are decoupled from [ships and aircraft].”[32] While disconnecting fleet-level capabilities from ship-level capabilities is valid for exposition purposes[33] and useful for exploratory studies, it would be erroneous to treat the two capability sets as totally independent.

A famous example of this is the Thach Weave maneuver developed during World War II [42] [43]. Well into 1942, Japanese fighter aircraft were aerodynamically superior to those of the USN [43]. This difference in performance meant US pilots were at a significant disadvantage against Japanese fighter pilots in one-on-one dogfights. This capability gap left ships vulnerable to torpedo bombers, which then impacted fleet-level capabilities. Supposing, anachronistically, the USN acquisition community had then performed a CBA, it would have evaluated at least two options: (1) fill the capability gap with better means in the form of more advanced fighter aircraft (which, over time, it did) or (2) develop better ways of using its current fighters. Had LCDR John S. Thach participated in the CBA, he would have proposed the combat maneuver[34] now named after him, one that dramatically increased USN fighter aircraft lethality and survivability[35] (even against overwhelming odds) and also rendered them capable of defending other ships and aircraft. Since Japanese aircraft were lighter and more maneuverable (neglecting armor for

[32] See [38] and its references.
[33] By analogy: The parts of an airplane cannot fly, but an airplane can fly. Here, “being thrown” is not considered flying.
[34] The new maneuver was a ways solution equivalent to new training and doctrine.
[35] Lethality and survivability are sometimes considered apart from other capabilities (for example [270]). Clearly, all military capabilities depend on these two, which is a simple example of how a fleet-level capability can be impacted (enabled/disabled) by a ship/aircraft-level capability. Deeper discussions begin in Sections 1.5-1.7.

critical regions such as the cockpit [44] [45]), the Thach Weave exploited weaknesses whose significance Japanese designers may have underestimated. The fact that both ways and means contribute to capabilities implies that as adversaries adapt to one-another, one side's unforeseen vulnerabilities (in ways or means) can inspire the other side's capability development. Therefore, it is paramount for the acquisition community to have a traceable approach for defining and modeling capabilities as well as a rigorous approach to mining modeling and simulation data for exploitable patterns of ship and aircraft interactions.[36]

Current system decomposition methods routinely capture intended/anticipated system interactions [46] [47]. These methods generally fall into two categories: those that decompose capabilities[37] into smaller tasks / system behaviors into subsystem behaviors and interactions, are generally known as ***functional decomposition***, while those that decompose systems based on their physical assembly are referred to as a ***physical decomposition***.[38] Examples of such methods include Rapid Architecture Alternative Modeling (RAAM) [48], the Interactive Reconfigurable Matrix of Alternatives (IRMA) [46], and several other approaches characterized using a Design Structure Matrix (DSM) [49]. Most assume a hierarchical structure[39] in which one 'higher level' behavior (i.e. system) is a consequence of 'lower level' behaviors (i.e. subsystem), and many can be partially or completely represented using a matrix, and, therefore, a graph [50].

---

[36] An experiment based on the Thach Weave (incorporating ideas by John R. Boyd) were added to this thesis in response to feedback received during the proposal. See CHAPTER 5 and CHAPTER 7.
[37] A civilian analogy to "capabilities" is the objective tree which decomposes goals into objectives, and then uses objectives as the basis for a functional decomposition. See Sections 3.5 and 3.7 of [47].
[38] Physical decomposition is just one of many terms that can be found in literature including *component-based decomposition* [49], or *system architecture* [46].
[39] Heterarchical structures are well known in biology, and are often discussed in the literature on emergence (Section 1.7). The interested reader can also refer to Section 4.2 of [272] for a brief, classic example of a heterarchical structure.

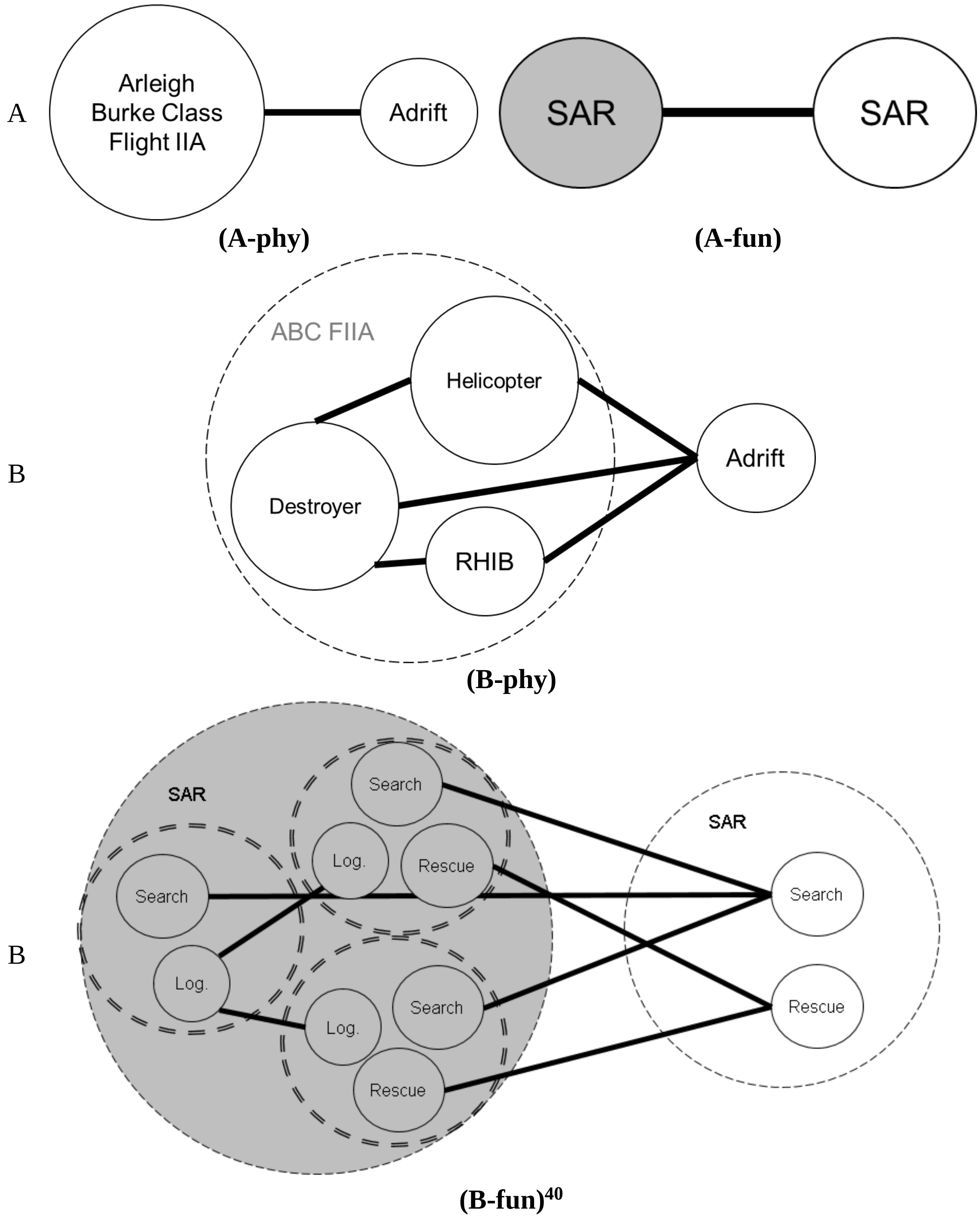


Figure 2 –A notional system decomposition for an Arleigh-Burke Class Flight IIA destroyer performing a Search and Rescue (SAR) mission.

---

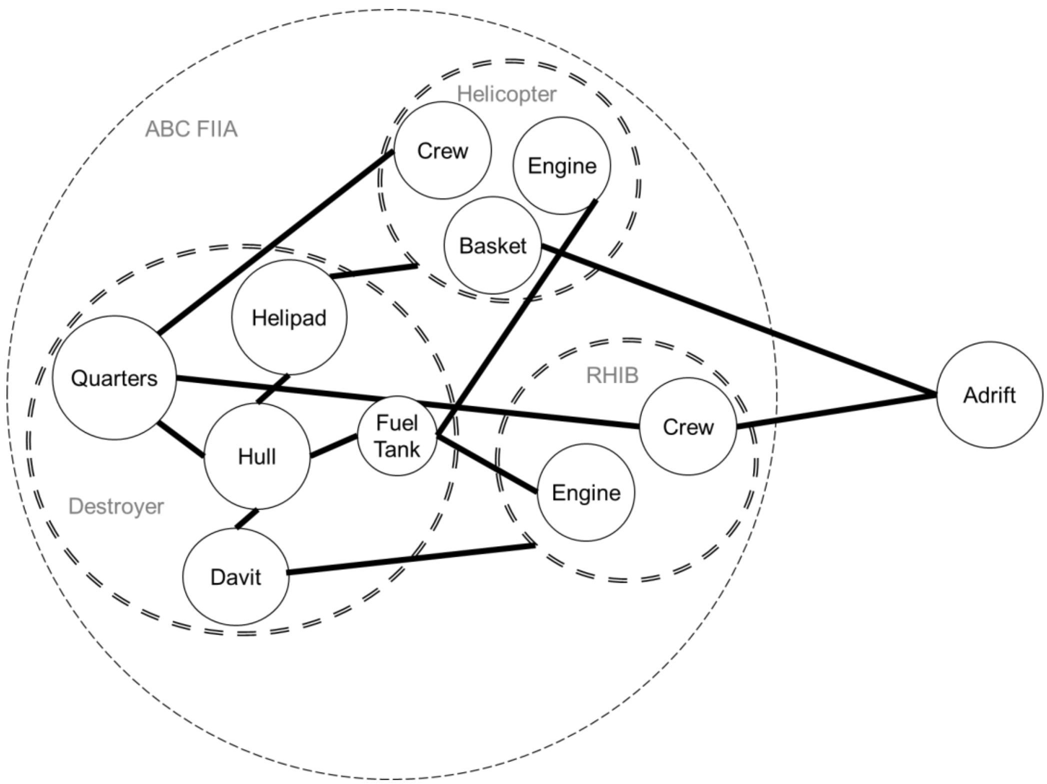
[40] Terms specific to Figure 2 and Figure 3 provided for clarity, but not otherwise covered: (1) "Log." = logistics; (2) "Comm." = communication; (3) "Loc." = locate; (4) "X-port" = transport.

Consider, then, the graphical representation of the system decomposition for a notional search and rescue capability depicted in Figure 2-Figure 3. The figures contain a notional physical decomposition (phys), and functional decomposition (fun) for an Arleigh-Burke Class Flight IIA (ABC FIIA) destroyer performing a Search and Rescue (SAR) mission to rescue the crew of a boat that is adrift. The figures show the three possible levels of abstraction, where A is the 'highest' level (least detailed), and C is the 'lowest' level (most detailed). Physical systems and functions are depicted as nodes in their respective graphs, connected by edges (solid lines) that represent (phys) interactions, or (fun) dependencies. A gray node 'supplies' the function that is 'requested' by a clear node. Although information-sharing behaviors (locate, communicate) are shown, their corresponding physical (sub)systems are omitted from the (phys) graphs. The first observation to make is that the adrift crew (Adrift) does not require further physical decomposition to convey its essential interactions, but its functional composition certainly requires more detail since merely stating "SAR" says nothing about the tasks required to adequately represent the 'ABC FIIA – Adrift' interaction.[41] However, in order to provide a more detailed description of the tasks 'requested' by the Adrift node (the need to be located, the need to be lifted out of the water or disabled boat, and the time-sensitive need for shelter), the ABC FIIA must be decomposed both physically and functionally since its SAR capability is a composite of the capabilities of its constituent subsystems. For example, we see at Level B that although the destroyer can help search for the Adrift, it is

[41] Since the 'Adrift' node is meaningful at multiple levels of abstraction/detail, this is an example of one type of heterarchical structure. See Footnote 39.

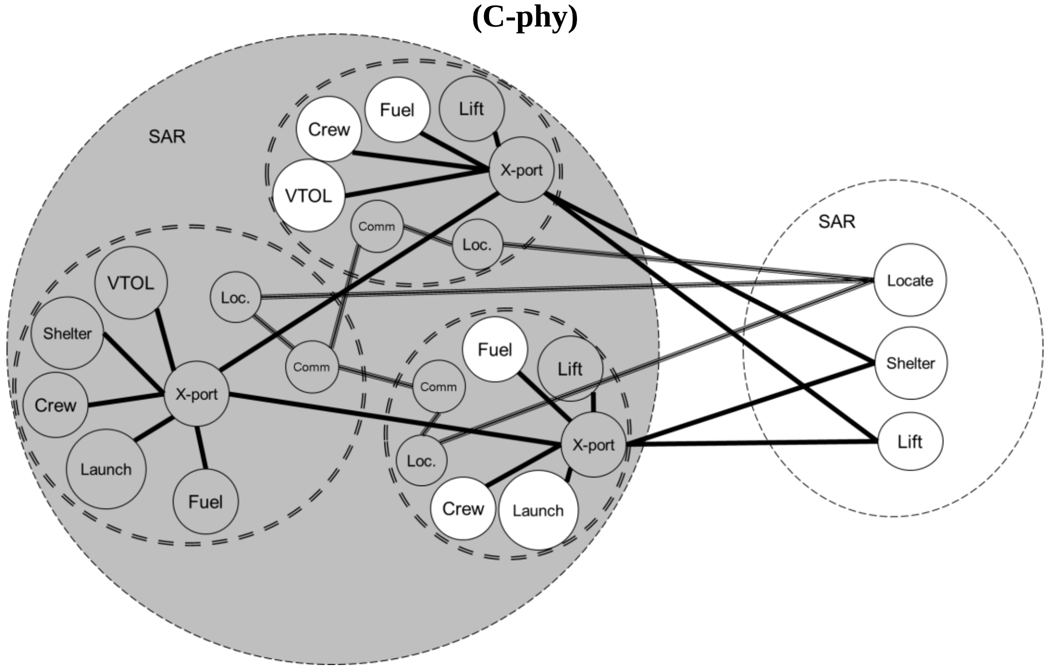
not equipped to physically rescue them, and relies on a helicopter or rigid-hull inflatable boat (RHIB) to perform those tasks.

C



(C-phy)

C



(C-fun)

Figure 3 – Continuation of the notional system decomposition from Figure 2.

Level C then depicts the various ways systems rely on each other (communicating location information, providing fuel or crew to pilot the systems, etc.), thereby showing that the nodes of each graph at each level of abstraction can be decomposed into collections of interconnected nodes, which are graphs themselves.[42] That is, a 'single interaction' at one level of abstraction may in fact represent myriad interactions at some lower level of abstraction, all of which have individual requirements that must be met in order for the 'higher level behavior' to be observed.

The critical reader will note that "search and rescue" is a compound phrase. This 'capability' was deliberately selected as a convenient and hopefully anodyne rhetorical device to illustrate that a 'single' fleet-level capability can be the consequence of multiple ship- and aircraft-level capabilities somehow combined. This immediately raises three questions. First, "Are system decompositions like those shown in Figure 2 - Figure 3 unique?" to which the answer is generally "no," particularly since the mapping from physical form to function is not one-to-one.[43,44] The second, stronger question is, "What about cases where a higher-level capability is not a simple combination of two or more lower-level capabilities?" Finally, the third, and strongest, question becomes, "How can one discover an unknown higher-level capability by building relationships up from some

[42] In addition nodes of nodes, and graphs of graphs (called ***hypergraphs***), the edges become multi-edges, and in more sophisticated representations, would result in directed, multi-edge hypergraphs. Similar observations have been made in works on interoperability such as the oft-cited work by Major Thomas C. Ford [26].
[43] Borrowing a concept from Mathematical analysis, two objects are ***one-to-one*** if and only if they uniquely correspond to one another. Thus, a function and a physical form are one-to-one if only that one physical object can perform that function, and vice-versa. For example, 'flying' is a function that is not one-to-one with form (bats, birds, airplanes, helicopters, etc.).
[44] Some authors capitalize on this when studying the ability of architectures to perform capabilities using interchangeable systems/functions. For example, the Engagement Generation Matrix in Figure 20 of [81], and the Capabilities and Requirements section of [67]. This claim can also be inferred from the fact that SoS requirements generally fail to have a one-to-one correspondence with system requirements [67].

initial lower level?" Alternatively, the third question could be phrased, "Is it possible to decompose a high-level observed behavior that is not a simple aggregate of two or more lower-level behaviors?" The next two sections will show that although one can 'readily' reduce a system to its physical parts [51], one cannot easily reconstruct the behaviors of those systems from the behavior of their parts, particularly when that 'system' is a system of systems (SoS).

1.5 System of Systems

Much of the literature on SoS is rather emphatic that the profession of SoSE should not be equated with Systems Engineering (SE). For example,[45] Pratt and Cook state, "SoSE is inherently a socio-technical activity and to succeed substantial effort needs to be dedicated to the social, cultural, political and enterprise aspects of the SoS and its engineering" [52]. While these distinctions play a role in the practice of engineering, they do not all extend to the study of systems science, which focuses on concepts that apply to any system implementation.[46] For example, consider the author's use of the terms social, cultural, political, and enterprise. Each of these terms denote different levels of abstraction, relevant to different hierarchies created from compositions of elements defined within the context of different fields of expertise.[47,48] Clearly, the study of hierarchies applies whether the hierarchy is political, social, etc. Furthermore, in each case, the argument stands that the behavior of the group cannot be easily reconstructed from the behavior of their parts.

---

[45] The remaining arguments raised in [52] concern how the professions ought to be structured, which is outside the scope of this work. For more information see also [64].
[46] See Figure 2.8 of [53] for a graphical depiction of systems science and its scope.
[47] Consider an explicit argument from sociology [277], which begins by quoting Emile Durkheim: "There can be no sociology unless societies exist… societies cannot exist if there are only individuals." A similar argument could be made for International Relations, Religious Studies, Darwinian Evolution, Political Science, Set Theory, Chemistry, etc.
[48] This is reminiscent of the way in which the distinctions between Chemistry, Electrical Engineering, Mechanical Engineering, and Medicine ultimately led to the creation of Biomedical Engineering as a field in its own right.

In order to focus on that reconstruction, this research requires a systems science-style definition of SoS that holds for any hierarchy, any composition of elements, and any level of abstraction.[49]

One solution to this is a "model-based" definition [46] of the term ***system***: "A system is a combination of interacting elements integrated to realize properties, behaviors, and capabilities that achieve one or more stated purpose(s)" [46]. This definition has the advantage of being broad enough to capture any combination of elements at any level of abstraction within the discipline of engineering.[50] The hierarchical nature of the system (i.e. the fact that it contains two simultaneously coexisting levels of abstraction) is inferred from the fact that the elements are "integrated to realize properties..." that would not otherwise exist. Therefore, the properties, etc., of the elements are distinct from those of the system. The model-based definition is also intuitive. In Section 2.1 of the 2015 ***International Council on Systems Engineering*** (INCOSE) SE Handbook[51] (the section devoted to defining the term ***system***) it reiterates that its various terminology essentially elaborates on "the fundamental idea that a system is a purposeful whole that consists of interacting parts" [53].[50] Finally, the model-based definition can be applied recursively, enabling a hierarchy of abstraction that extends beyond two levels. Therefore, applying the model-based definition once again, a ***system of systems*** can be defined as "a combination of interacting systems [i.e., elements of the SoS] integrated to realize properties, behaviors,[52] and capabilities that achieve one or more stated purpose(s)" [46].

Recently published primary literature uses definitions very similar to the model-based definition when introducing SoS. For example, compare the graphical representation

---

[49] Something of an echo of General System Theory [279].

[50] Outside of engineering, one might prefer a minimalistic concept of system achieved by dropping the "purpose" qualification. For example, the formation of molecules by atoms is coupled to an energy exchange, and occurs whether or not the resulting molecule, or energy exchange, serves any particular purpose.

[51] Going forward, this will simply be referred to as the SE Handbook.

[52] The terms ***properties*** and ***behaviors*** will properly defined in Section 2.3.

of a system provided by the SE Handbook (see Figure 4) to the definition of SoS in the 2018 INCOSE SoS Primer:[53] “a collection of independent systems, integrated into a larger system that delivers unique capabilities. The independent constituent systems collaborate to produce global behavior that they cannot produce alone.” [54] In this sense, one can say, for example, that a spark plug is to electrical current, as an internal combustion engine is to torque, as a car is to locomotion. As simple as it may seem to extend the model-based definition, it must be noted that even as recently as 2011 there was no broad consensus on how a SoS ought to be defined [55] [56],[54] largely due to the various professional considerations required for SoSE.

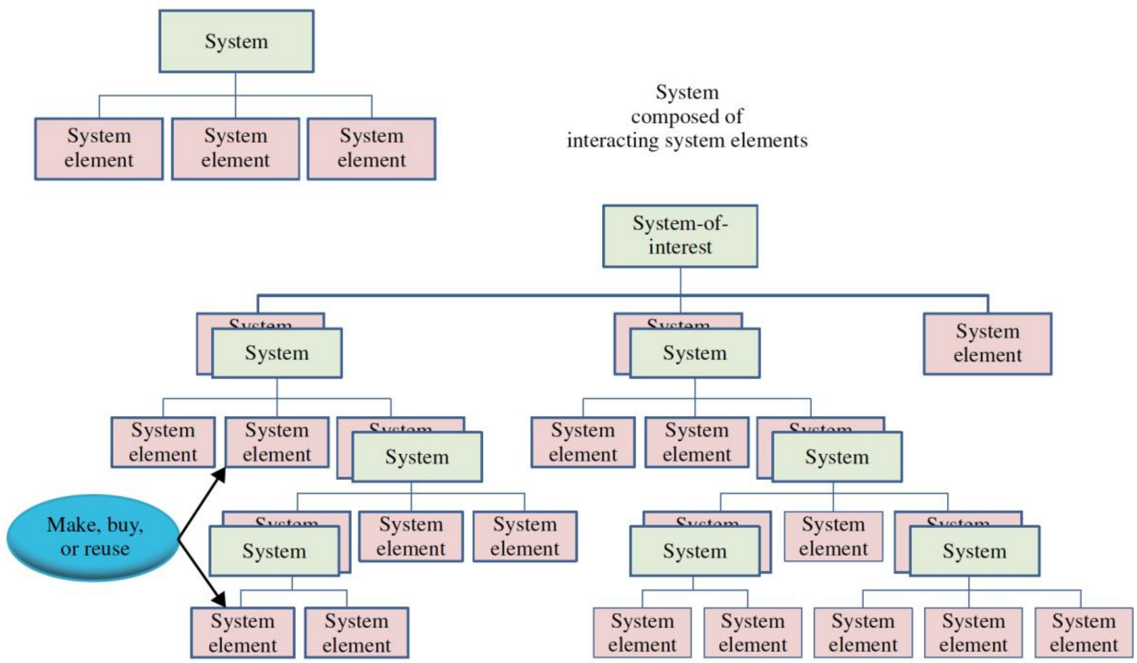


FIGURE 2.1 Hierarchy within a system. This figure is adapted from ISO/IEC/IEEE 15288:2015, Figure 1 on page 11 and Figure 2 on page 12, with permission from the ANSI on behalf of the ISO. © ISO 2015. All rights reserved.

Figure 4 – Graphical depiction of the term *system* reproduced from [53].[55]

[53] Going forward, this will simply be referred to as the SoS Primer.
[54] What appears now to be the trend was characterized in 2009 as “an opinion among many systems engineers and other stakeholders.” See page 37 of [46].
[55] Notice that systems can interact with other systems and system elements at various levels of hierarchical abstraction. This is another example of a heterarchical structure discussed in Footnotes 38, and 41. Here a system is a collection, and can also be part of another collection, and can interact with the elements of a collection. The element/system hierarchy is not strictly vertical.

The simplicity and convenience of the model-based definitions, particularly when taken out of context, belie the challenge inherent in distinguishing between a system and a SoS. This distinction must be clear in order to meaningfully attribute behavior to the SoS.

Previous attempts at distinguishing between the systems and SoS tend to emphasize concerns relevant to the organizations they operate in. The SE Handbook references Maier's five characteristic that distinguish a system from a SoS: (1) operational independence of constituent systems, (2) managerial independence of constituent systems, (3) geographical distribution, (4) emergent behavior, and (5) evolutionary development processes [53]. Three of the five are clearly organizational, and therefore, beyond the scope of this work. In this context, "geographical distribution" is a logistical concern relevant to many organizations (for an example, see [57]).

Systems tend to…	Systems of systems tend to…
Have a clear set of stakeholders	Have multiple levels of stakeholders with mixed and possibly competing interests
Have clear objectives and purpose	Have multiple, and possibly contradictory, objectives and purpose
Have a clear management structure and clear accountabilities	Have disparate management structure with no clear accountability
Have clear operational priorities, with escalation to resolve priorities	Have multiple, and sometimes different, operational priorities with no clear escalation routes
Have a single lifecycle	Have multiple lifecycles with elements being implemented asynchronously
Have clear ownership with the ability to move resources between elements	Have multiple owners making individual resourcing decisions

Figure 5 – Various distinctions between systems and SoS, reproduced from [54]

However, it also conveys the sense that a SoS operates over a larger scale than a system, a concept that will be revisited in later sections.[56] The term "emergent behavior"[56] suggests a new kind of behavior not present in ordinary systems. The SoS Primer provides a table of distinctions, depicted in Figure 5. One point that stands out in Figure 5 (2nd row) is the assertion that, broadly speaking, a system's properties, behaviors, and capabilities are precisely what they were designed to be due, in part, to there being more specific design objectives and far fewer decision makers relative to SoS. Thus, somewhat vaguely speaking, systems are expected to be simpler. The guarded qualification "tends to" reveals that this is not always the case in practice.

SoS, themselves, are categorized by their organizational structure. There are four canonical types [54]. ***Directed SoS*** are centrally managed/operated, and although their constituent systems operate independently, the central authority dictates their normal operation. ***Acknowledged SoS*** have a central authority, but changes to operation are decided collaboratively. ***Collaborative SoS*** have no central authority, and changes are decided collaboratively. ***Virtual SoS*** have no central organization or goal. Of the four, only the Directed SoS is defined as being "built and managed to fulfill a specific purposes"[57] [54]. This category of SoS explicitly blurs the distinction between a system and a SoS (the others rely on organizational/operational distinctions). Nevertheless, on the whole, engineers expect there to be a significant distinction between the simplicity and behavior of a system and a SoS.

[56] See Section 1.7 and CHAPTER 4. Nearly every reference on SoS in this thesis notes that they display emergent behavior.
[57] Using this terminology, Navy Fleet Synthesis studies are studies of a Directed SoS.

Using a synthesis / definition reconciliation approach in her PhD Thesis, Dr. Griendling writes, "commonalities between the definition of complex system and SoS implies that while a complex system is not necessarily an SoS, an SoS is almost always a complex system" [56]. Once again, a SoS bears a strong resemblance to a system; specifically, a "complex system." It follows that one common approach to distinguishing between a system and SoS relies on defining and quantifying a property called ***complexity***. Unfortunately, complexity has multiple definitions in many fields. This thesis requires borrowing the definition from Computer Science (CS) rather than SE in order to associate the model-based definition of ***system*** with the mathematical/computational calculation of ***model complexity*** (see Sections 2.2 and 5.1.3).[58] The remainder of this discussion will utilize complexity in the SE sense. For the purposes of this thesis, a ***complicated system*** is an incredibly intricate system whose behavior is nevertheless well-understood (such as an automobile) [53]. Readers interested in extensive reviews of the SE definitions as well as a rigorous disambiguation of the terms ***complex*** and ***complicated*** are referred to [56] [53] [58] [59] [60] and their references.

1.6 Complex Behavior

Like the term SoS, there has been extensive discussion (and disagreement) on how the term complexity ought to be defined.[59] As recently as 2015, the Chair of the INCOSE Complex Systems Working Group, Dr. Jimmie McEver, stated that there is no "easy, agreed-upon definition" for complexity [61], a point reiterated in 2018 by Computer

[58] As will be seen in the following sections, SE uses complexity in a manner often interchangeable with emergence, whereas CS uses it solely to quantify the difficulty of evaluating an expression or executing an algorithm. This thesis will use the term emergence to refer to a quality that is modeled (and a behavior that can be modeled), while complexity will be used for a quantity that is measured, which is closer to CS.
[59] Thirty-three different types of complexity are listed in [278], some from different branches of science.

Scientist Dr. Russ Abbott [62]. Despite this disagreement, a variety of sources contain useful, and largely compatible, lists of generally well-defined characteristics that contribute to system complexity [63] [64]. Since most of these factors are easily relatable, it suffices here to briefly highlight these factors. One list by McEver relates system complexity factors to the challenges faced by decision-makers and system engineers (Figure 6):

Hallmarks of complexity	**Impact on Decision Maker**
Interdependence	Cannot treat by decomposition
Nonlinearities	Extrapolation of current conditions → error
Open boundaries	Cannot focus only on processes inside boundary
Multi-scalarity	Have to address all relevant scales
Causal & influence networks	Challenge: develop 'requisite' conceptual model within time and information resource constraints
Emergence	Unknown risks and unrecognised opportunities
Complex goals	Goals may change, be unrealistic, vague
Adaptation & innovation	'Rules' change, interventions stimulate adaptation
Opaqueness	Many possible hypotheses about causal paths, insufficient evidence to discriminate

Figure 6 – The impact of system complexity factors on decision making, reproduced from [61].

A key distinction between a standard system and a ***complex system***[60] (i.e. a system exhibiting ***complex behavior***) is that its behavior cannot be fully understood by examining the system's parts in isolation [53], meaning that it cannot be unambiguously characterized using standard decomposition (Figure 6). As McEver puts it, "The opposite of 'complex' is 'decomposable', not 'simple'" [61]. The same distinction is frequently made for

[60] Dr. Balestrini-Robinson presents a straightforward taxonomy (see Figure 10 of [81]) wherein a system's complexity can be designated using the combined complexities of its physical and functional decompositions, provided that those complexities can be adequately determined. He also provides an extensive list of complexity characteristics in Section 2.1 of his thesis. McEver presents another straightforward taxonomy, associating systems of varying complexity with Cynefin domains [60].

emergent behavior (see also [65] [66] and Section 1.7). Referring again to the 2015 SE Handbook, "The SoS usually exhibits complex behaviors, often created by the existence of the aforementioned Maier's characteristics… In complex systems… interactions between the parts exhibit self-organization where local interactions give rise to novel, nonlocal, emergent patterns..." [53]. As before, this complex behavior (this time referred to as novel behavior associated with emergent patterns[56]) is one that is assigned to the system level of abstraction, and cannot be generated by any one system element. That is, element interactions can somehow generate complex system behavior, system interactions somehow generate complex SoS behavior.[61] This is the "multi-scalarity" of complexity referred to in Figure 6, wherein information that is relevant at one level of abstraction is either obscured or irrelevant at another level [67].[62] Although it is possible to create a hierarchy of behaviors, NASA administrator Michael Griffin acknowledged[63] that "complex systems are no longer strictly decomposable, and systems engineering has to adapt" [60]. McEver further states that this adaptation cannot be achieved by merely extending standard SE techniques [61]. Entirely new approaches are needed for associating lower-level interactions with higher-level behaviors because it is not always obvious which interactions cause which behavior ("opaqueness" in Figure 6). A 2014 survey of INCOSE SoSEs designed to help inform changes to SE relevant to SoS studies similarly concluded, "The inability to predict SoS behavior, especially when the constituent systems themselves are complex systems, is an area of risk for SoS" [68], and cites an unnamed respondent that

[61] Once more for clarity: system *interactions* generate *SoS behavior*, which must not be confused with saying that a system's independent behavior generates SoS behavior (as though a SoS directly appropriates SoS behavior from systems).
[62] Note that Figure 6 is McEver's summary of the long-form discussion in [66].
[63] The citation is dated 2010. The quote itself may be older.

said, “well-structured approaches for 'design for emergence' are not generally available” [68]. Despite some attempts to the contrary (e.g. [69]), the 2016 INCOSE Complexity Primer[64] calls for abandoning the notion that a complex system can be designed or controlled, and instead encourages thinking in terms of “‘influence’ and ‘intervention’” [70], adding that “designing or evolving a complex system requires recognition that the designer may not ever be able to control or even understand the system completely” [70].

Consider the straight-forward example of a stair-climbing machine presented in Figure 7.

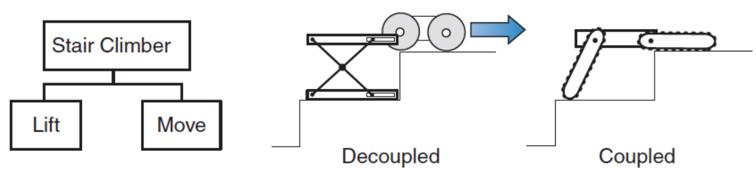


FIGURE 20.1 Two Concepts for a Stair-Climbing Machine. The first concept (center) decouples the lift and move. Although it still does the job, the second concept (right) does not.

Figure 7 – Stair-Climbing Machine Concepts in [71]

A standard functional decomposition is one where the systems engineer can draw a graph showing how independent, lower-level functions contribute to a higher-level function.[65] As the original caption in Figure 7 states, the lift and move functions are independent because no interaction between the wheels and elevator is needed to perform those functions. A coupled system, on the other hand, requires a graph containing cycles (as shown in Figure 8), and thus is not decomposable in the traditional sense. Before elaborating on the meaning of decomposition in the “traditional sense,” note that the

[64] Going forward, simply the Complexity Primer.
[65] See Chapter 3 of Kitto’s thesis for an extensive discussion on hierarchies and decompositions [73].

convention of the diagram used in [71] is analogous to the hypergraph discussed in Section 1.4 (see Figure 2-Figure 3).

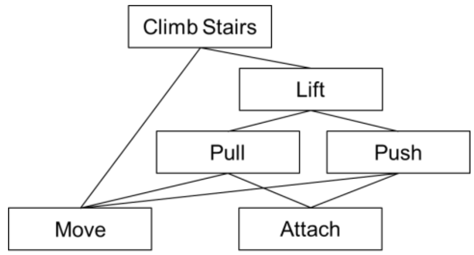


Figure 8 – Suggested Functional Decomposition for Coupled Stair-Climber

The vertical distribution of nodes in Figure 7-Figure 8 corresponds to the level of abstraction of the function they represent (higher up is a higher level). These types of graphs are called ***layered graphs***. One could argue that cycles will appear in any graph where the structural-functional mapping is not one-to-one,[66] but the key here is that certain interactions create functions that could not otherwise exist (the motion of the track connected to the rigid bar creates the push/pull that ultimately enables the climbing behavior). The cycles of the layered graph, or multiple levels of the hypergraph suggest why standard decompositions break down, and why SoS primer speaks of influencing behavior rather than controlling it. The underlying logic of the functional decomposition is that the outermost nodes are the independent functions that can be directly manipulated to achieve the higher-level function, and that this can be done recursively as the system becomes more complicated (every independent node can be further subdivided into more independent nodes). However, in the case of complexity, it is the intermediate, coupled functions that most directly cause higher level behavior. Unfortunately, those intermediate

[66] Examples of structure-function mappings that are not one-to-one are shown in Figure 6 of [83].

functions cannot be directly controlled,[67] and the more cycles there are in the graph, the harder it becomes to simultaneously balance low level functions in order to achieve the higher level function.

<u>Gap</u>: Standard functional decomposition methods do not extend to complex behaviors in the sense that independent functions at the extremes of the graph (those that can be *directly controlled*) no longer correspond to the coupled functions that *directly cause* the desired (complex) behavior.[68]

Readers familiar with graphs will immediately note that layered graphs are, in fact, directed graphs, and that the cycles discussed above do not satisfy the definition of a cycle in a directed graph. In the layered graph provided above, all edges can be redrawn as arrows pointing up. In order for a directed graph to have a cycle, there must be some kind of feedback loop (the higher level node must have an arrow pointing back down at a lower level node). Feedback loops will be briefly discussed in Section 1.7, but are largely outside the scope of this thesis. This thesis will only go so far as to say that a complex system can be represented using a functional decomposition who's layered, undirected graph contains cycles (i.e. coupled functions). This will be treated as a necessary condition (see again Section 1.7), but will remain a topic for future study.

Although these factors do not create a hard distinction between system and SoS, it is clear that an element/system/SoS hierarchy compatible with a model-based definition

[67] This is where an engineer might be tempted to insert a simplifying assumption that would enable them to draw a graph where the complex function is represented by a single node at the bottom, and the coupled functions are hidden from view. This simplified graph would give the illusion that the system is not complex.
[68] A hopefully useful aphorism: you no longer have direct control over the direct cause because the true cause is indirect.

can be created, in principle, by identifying those behaviors that can only be attributed to the higher level of abstraction. There are at least four reasons a generalizable approach to this kind of hypergraph or layered graph creation has proven so elusive: First, the basic premise that properties can/ought-to be assigned to an object is "more a matter of describing how we think… than a matter of characterizing nature" [72]. Our brains are generally thought to be built for pattern recognition and possess the ability to associate abstract entities with one-another. Therefore, we naturally seek to categorize the entities we perceive despite the fact that those entities persist and interact without regard for such categorizations. Furthermore, we can only successfully create hierarchies of objects for whom our way of thinking is compatible with nature. Entities that cannot be somehow associated with stable patterns are difficult to understand. Second, utility is subjective, and complexity is creative. Without a complete, perfect understanding of the environment in which the complex system will operate, it is impossible to conceive of every way the properties and behaviors of the complex system and its parts will be observed or capitalized on by some other entity and vice versa. In other words, there is no way to predict every possible higher level interaction without some knowledge of what other objects exist at that higher level. If that were not the case, modeling the evolutionary tree of life on Earth backward or forward in time would be trivial. Similar observations appear throughout the SoS literature (e.g. see discussion of contextual systems in Section 1.3 of [73]). Thirdly, systems science can only ever classify an object, in general, as an "…of systems of systems of …"[69] Every gadget can be decomposed into parts, and then into materials, and then into molecules, and so forth. What lies at the origin of the sequence? Atoms gave way to quarks,

[69] Similar to the cursory "matter of perspective" observation made in Section 1.4.1 of [56].

and then possibly to strings (atoms are 'obviously' the consequence of interacting quarks and some argue that quarks are the consequence of interacting strings).[70,71] For the systems scientist the question becomes, how does the sequence end, if it ends? Worse still, even if such an end could be reached, a complete, exact model of nearly any system would be impractical. Therefore, system engineers never directly control (i.e. interact with) every individual basic element of the system in question, nor do they operate with a complete picture of the context in which the system exists. Fourthly, many scientists and engineers tend to over rely on philosophical Reductionism (not to be confused with Reductive Analysis [73]) in their research and analyses (for arguments see [74] [75] [76] [77][72]). Greedy Reductionism [73], in particular, presents the engineer with a fundamentally self-defeating paradox (it is inherently dehumanizing). By claiming that an object is no more than a collection of parts, this philosophy essentially argues that one can best practice Systems Engineering by first assuming that the 'systems engineer' does not exist.[73] More importantly, it creates a bias in the minds of scientists and engineers, which then further impedes their ability to create hierarchies of complex behaviors. See [78] [79] for an examples of this in medicine. Readers interested in long-form arguments are encouraged to review the aforementioned references.

Acknowledging these challenges, various studies have proposed metrics for quantifying the complexity of a system/SoS, based on carefully selected SE definitions of

---

[70] Some physicists argue there can be no particle more fundamental than quarks. See Section 2.5 of [108]. This argument is just for illustration.

[71] Philosopher and smart person Emily Levine interweaves this subject into her TED talk in a way that adds richer perspective to this discussion [296]. Thank you Emily.

[72] Explicit examples of the enduring appeal of Reductionism (including the hope of identifying the "Theory of Everything" argued against by Laughlin [75]) are available in Chapters 2 and 8 of [108].

[73] Rowan's paper on "reductive logic" traces this line of reasoning back to Descartes. He argues one major problem of this kind of logic is a "self-referencing paradox" summarized by the aphorism: "I think reductively, therefore I am not" [74].

complexity. In an extensive review of complexity measures, Dr. Witold Kinsner, former President of IEEE Canada, states, “Complexity appears to be context sensitive, and cannot be defined universally, once and for all” [80]. Although there is truth to this statement, Kinsner’s review covers multiple scientific disciplines, which makes it impossible to define complexity universally. Within SE and SoSE, there is no reason to think some useful consensus on complexity is unobtainable. One common approach to measuring complexity is to use metrics derived from the graphs of the system/SoS physical or functional decomposition [56] [81] [60] [82] [83] [76] [58]. Metrics calculated from standard (decoupled) functional decompositions do not apply here. Metrics calculated from physical decompositions will confound complexity with complicatedness, making them unreliable. This leaves metrics calculated from graphs of layered, cyclic functional decompositions or hypergraphs. Of the aforementioned references that implement these metrics, none of them perform calculations on hypergraphs. Again within that subset, only Flanigan utilizes a cyclical, directed graph of a functional decomposition (see Figure 16 of [83]), but that graph depicts a single level of abstraction and no complexity measurements are taken using that graph (Flanigan’s graph-theoretic complexity measures are performed on structural decompositions, as in Figure 19 of [83]). A rigorous study of graph-theoretic measures of complexity on functional decompositions represented using layered graphs or hypergraphs appears to be an open topic in the literature, and is outside the scope of this work. The approach to functional decomposition provided by Brimhall et al. [84], combined with the aforementioned metrics appears to be a good starting point for such a study.

Other methods are derived from Information Theory [85], such as calculating the information entropy [86] [87] [88] [86] [89] of a system to directly or indirectly measure complexity. There are also measures relying on thermodynamic entropy [90]. Many of these metrics are not one-to-one (e.g. the ratio measurement scale used in [60]). In such cases a SoS of several, decomposable systems can have complexity on par with a single, complicated-and-complex system, or with a SoS of few, complex systems. Therefore, it can be a leap to make an ontological/categorical argument from any particular value of a metric [76].

Still other methods are listed in the extensive, thoroughly referenced reviews by Kinsner [80] and Shalizi [91]. Most of the metrics reviewed by Kinsner are designed to measure forms of complexity outside the scope of this thesis (including structural, dynamic, synergetic, and design complexities [80]). However, Kinsner's "functional complexity" is precisely a discussion of complex behavior. Unfortunately, it is also the topic Kinsner writes about the least (devoting to the topic roughly 7 sentences and 2 references by the same author, within a 31 page paper). Kinsner mentions in passing that multiscale metrics can be used to measure functional complexity. Ay et al. created what they call a unifying framework for complexity measures, which covers multiple information entropy measures and is designed for hierarchies of interacting levels, would be directly applicable to several examples of finite systems [92]. Both Ay and Kinsner's multiscalar measure do appear to be applicable, since layered graphs and hypergraphs capture this multiscalarity, but are only unambiguous after the graph is drawn, which is to say after the behavior has been identified and named. Prior to that (for example, using a data set that contains undetected complex behavior), the metrics may show interesting mathematical features when the complex behavior is exhibited, but there is no obvious way to determine what that behavior is given the value of the metric. Shalizi's methods are closer in spirit to the ideas that will be adopted here, but they suffer the same drawbacks of those covered by Kinsner. In the absence of a rigorous method for functional decomposition (tied directly to mathematical models that predict the associated quantities), there is no clear cause-effect relationship between these metrics and *qualitatively identified* complex behaviors [67].[345] These metrics suggest the possible existence of complex behavior, without specifically identifying what that behavior is, how it is derived from system interactions, or how it should be modeled at the SoS level.

In his PhD Thesis, Dr. Domerçant discussed Kinsner's observations and concludes that "many of the existing complexity measures developed by complexity scientists tend to be very domain specific or too theoretically abstract to usefully apply to real world systems." He subsequently argues that "the overarching reason for this is that the diversity that exists among both natural and engineered systems makes it difficult at best to define an absolute measure of complexity that is applicable to any and all systems…" [60]. One benefit of the model-based definition of a system is that it does not make the distinction between engineered and natural systems (or, at least, it need not[74]). Furthermore, Kinsner appears to argue that complexity definitions are discipline-specific. Two excellent examples are the complexity of a graph in mathematics and the complexity of an algorithm in computer science, the latter of which will be explored in this thesis. Whatever challenges a system engineer may encounter in defining complexity for all systems, it appears that a lack of theory connecting quantifiable data to the qualitative identification of complexity leads to poor problem formulation methods that undermine the engineer's ability to clearly identify complex behavior.

Taking a decidedly practical tone, former director of the Research School of Systems Engineering at Loughborough University, Dr. R. Kalawsky describes the problem this way: "The inevitable complexity of today's products makes it difficult for a single individual to understand where the peaks in the product's performance lie against a landscape of different and often subtle design solutions where undesirable emergent behavior appears…" [93]. Dr. Kalawsky then goes on to list "Development of reliable early detection of undesirable emergent behaviour… especially for Systems of Systems" as one

[74] Reconciliation between natural and engineered systems can be achieved by adopting a non-anthropomorphic definition of "purposeful."

of the Grand Challenges in the Verification, Validation and Assurance of extremely complex systems [93]. The INCOSE "Systems Engineering Vision 2025," which communicates the top ambitions for the profession as a whole, also speaks to the need for "identifying emergent behaviors and dealing with unanticipated behaviors" [94]. INCOSE envisions a future where, by 2025, "standard measures of complexity will be established, and methods for tracking and handling complex system behaviors and mitigating undesired behaviors will be commonplace" [94]. Achieving this will require, "a shift in emphasis from reductionism to holism," adding that "Systems Science seeks to provide a common vocabulary (ontology), and general principles explaining the nature of complex systems" [94]. Clearly, the practical problems faced by engineers have their roots in the knowledge gaps confronting systems science. With these priorities in mind, the overarching problem for this thesis can be stated as follows,

Research Problem: The traditional SE approaches to defining the properties and behaviors of a SoS that are distinct from those of its constituent systems lacks generality and traceability, and results in designs whose behaviors are only partially understood, the remainder of which can be exploited for some unintended purpose.

In his paper on complexity and emergence, Philosopher of Science, Dr. Miguel Fuentes, observed, "There is somehow a strong connection, at least in a huge part of the community discussing emergent phenomena and emergent properties, between complex systems and emergence" [95]. Nevertheless, Dr. Vadim Kim observes in his PhD Thesis that, "There does not seem to be a clear connection between complexity *measures* and emergence. There is neither a *'critical' level* of complexity that yields emergence nor is there a problem-independent case to be made that *higher measures* of complexity yield emergence" (emphasis added) [59]. Since SE currently lacks methods for decomposing complex behavior, it is plausible that the lack of correlation between emergence and the measures studied by Kim is the result of researchers unknowingly measuring complicatedness rather than complexity, or taking measurements that confound complicatedness with complexity.[75] Furthermore, most references on complex systems or SoS within the SE/SoSE literature cited in this thesis explicitly associate complex systems with emergent behavior (including Kim).

1.7 Emergent Behavior

Supposing that 'search and rescue' is a behavior unto itself, it would be a perfect example of what an emergent behavior is ***not***: a simple series of vaguely contemporaneous behaviors [96]. Defining what an emergent behavior is, on the other hand, is much more controversial. As with SoS, and complexity, "there is no formal, universally agreed definition of emergence" [97]. This section aims to show the "strong connection" [95] between concepts that underlie complex behavior, emergent behavior, and SoS before stating the Research Objective of this thesis.

In the philosophical literature, the concept of emergence exists, in part, to explain causation. This thesis will study causation within the context of mathematical models and computer algorithms, wherein causality is unambiguously described (in principle). Readers interested in the broader philosophical discussion on causation are referred to [98] [99].[76] Two important types of causation discussed in the emergence literature are upward and downward causation. ***Upward causation*** is the anodyne, loosely defined idea that the

[75] At the end of Section 5.4.1 in [58], Kim states "These complexity measures only measure how much work it takes to describe or explain a process but does not actually capture the aspect we are most interested in: the system behavior." This is similar to the hyper-graph argument made earlier. See also CHAPTER 4.
[76] See the Appendices.

elements of a system contribute to, and thereby cause, the properties and behaviors of the system [100] [96] [101]. ***Downward causation*** is the much more contentious idea that the system causes changes in the properties of its elements directly, somehow, [96] [102] or by constraining them [101], or that a system interacts with other elements/systems directly[77] [100]. Both types of causation use the semantics of different levels, as in standard SE decomposition techniques. If downward causation were graphed using standard techniques, it might resemble a functional decomposition with feedback loops.[78] For example, see Figure 3 in [97]. It is not at all unreasonable to speak of feedback loops *within* a SoS given that, for system X contained in the SoS, the other systems within the SoS form part of the local environment of system X. The challenge arises when one endeavors to say that there is a feedback loop between a property of a SoS and a property of one of its systems. Nevertheless, if the number of systems is large enough, the interactions between system X and the other systems far outnumber interactions between system X and components of the environment external to the SoS, which would make the SoS the predominant influence (see similar arguments in [103]).

These forms of causation are also consistent with the model-based definition of SoS. That definition assumes that a SoS 'exists' with 'properties' and 'behaviors' that are distinct from, and generated by the behaviors/interactions of its constitutive systems. Relating this to causation, any SE can casually observe that 'governments collect taxes' and 'tax collection reduces individual short-term financial liquidity' as though government

[77] As opposed to saying the elements of system X interact with the elements of system Y, thereby making the interaction of X with Y somehow indirect or secondary. For example, the statement 'two people hug' can be thought of as an indirect cause-effect relationship, while 'atoms press against atoms' is the true, underlying, direct cause-effect statement (never mind the quarks, etc.).
[78] See [143] for discussion of feedback loops.

exists (a SoS with properties and behaviors upwardly caused by the citizenry) and is capable of directly affecting an individual citizen (downward causation) in a sense equivalent with physical object interaction.[79] These are examples of both the ‘multi-scalarity’ and ‘causal & influence networks’ inherent in complex systems (Figure 6). This leads to natural extensions, such as the hyperstructures presented by Baas [104], which are directly analogous to Systems of SoS, and hypergraphs (see Appendix).

The challenge in philosophy, as in SE [94], is to develop a logical approach for constructing a hierarchy that illustrates the direction of causation across/among levels as well as a foundation for assertions that some object exists.[72] In philosophy, this is referred to as the development of an ***ontology*** [105]. As will be shown later in the document, multiple scientists and engineers have endeavored to create an ontology for complex systems and emergent behaviors but did not refer to their work as such (it is an uncommon term in engineering). Since the scope of this thesis remains within the ‘comfortable’ limitations of mathematical and computational models, it is worth briefly revisiting the role reductionism has played within physics, where the identification of elements and levels is fairly uncontroversial and the predictive power of the current mathematical models is well established. In that field, bridging atomic-level properties or theories up to macroscopic properties of engineering materials is often achieved by[80] (1) assuming idealized particles with few or no interactions,[81] and/or (2) eliminating inhomogeneity by assuming intervals of time or space large enough for stability to be achieved, usually in the form of taking

[79] Readers accustomed to perceiving a hard distinction between the existence of objects like quarks, atoms, solar systems, a child tethering a kite, a government, or a culture are encouraged to read [51], as well as the various arguments attributed to Albert Einstein in [108] (see also Footnote 72).
[80] A similar argument is made in [115].
[81] For example, relating Kinetic Theory to the Navier-Stokes Equations [292].

limits,[82] and/or (3) using a patchwork of empirical, semi-empirical and/or theoretical models whose prediction range overlaps,[83] and then mathematically smoothing the often conflicting results of these coupled/interwoven models using a combination of mathematically plausible functions and expert judgment.[84] Although these methods are general, none of the resulting solutions generalize: (1) all simplifying assumptions are case-specific by definition, (2) what was once believed to be a micro/macro dichotomy has become a list of several levels complete with mathematical models for each level,[85] and a hierarchy of mechanical, electrical, thermal properties, etc., that operate at overlapping scales (3) all (semi)empirical models are specific to some controlled, laboratory or manufacturing setting, and all expert judgment is shaped by limited experience, at best, and cognitive bias at worst [106] [107]. When struck by the difficulties in traversing levels, some physicists operating from a reductionist philosophy explain that the difficulties are merely due to the sheer number of calculations required to predict higher level properties from lower-level data, and/or a mere lack of initial data ([108], for example, see the caption of Figure 2.1 in [109]). However, physicist and Nobel Laureate, Dr. Philip Anderson counters that this is a misuse of reductionism:

> *The main fallacy in this kind of thinking is that the reductionist hypothesis does not by any means imply a "constructionist" one: The ability to reduce everything to simple fundamental laws does not imply the ability to start from those laws and reconstruct the universe… The behavior of large and complex aggregates of elementary particles, it turns out, is not to be understood in terms of a simple extrapolation of the properties of a few particles. Instead, at each level of complexity entirely new properties appear… at each stage entirely new laws, concepts and generalizations are*

[82] As is the case for classical thermodynamics and continuum mechanics (see also discussion in [107]).
[83] See Figure 1.1 of [108]. See also [294]
[84] For example, see the discussion of "law of the wall" for turbulent flows [293]. Nonlinear perturbations of simpler models are also commonly used.
[85] In *materials science*, the *length scales* are electronic, atomistic, microscopic, mesoscopic, and continuum [108], never mind engineering, geology, or astrophysics.

> *necessary, requiring inspiration and creativity to just as great a degree as in the previous one. Psychology is not applied biology, nor is biology applied chemistry.* [110]

He later goes on to give the example of determining the shape of the atomic nucleus (loosely speaking), saying "Starting with the fundamental laws and a computer, we would have to do two impossible things – solve a problem with infinitely many bodies, and then apply the result to a finite system – before we synthesized the behavior." [110] Although physicists do not often use the terminology of emergence,[86] Anderson then uses the example of a crystal to highlight that, as the number of atoms increase, their organization forces new kinds of behavior to dominate the atoms in system that would not be obvious given the laws that govern their behavior individually.[87] He ultimately says, "…the whole becomes not only more than but very different from the sum of its parts" [110]. Within physics, these abstractions are closely associated with length or time scales, and thus, to a physicist, bridging levels of abstraction is equivalent to traversing length or time scales. However, as Philosopher Robert Batterman argues from the history behind the derivation of the Navier-Stokes Equations, "It seems that here may very well be a case where a continuum point of view is actually required: Bottom up derivation from atomistic hypotheses about the nature of elastic solid bodies fails to yield correct equations governing the macroscopic behavior of those bodies" [108]. The challenges faced by practitioners in a field as old and successful as physics make it clear that having a set of empirically

---

[86] Physicists prefer terms like 'multiscale,' [295] 'states,' or 'phases' to distinguish between abstractions. For example, one U.S. Department of Energy report on multi-scale physics [282] uses 'emergent behavior' and 'emergence' in a more layman sense.
[87] In the literature on emergence, this phenomenon would fall under the general category of ***self-organization***, which is discussed in CHAPTER 3.

validated equations that characterize the behavior of the elements of a system is only one step toward characterizing the behavior of a system.

Returning to the literature on emergence, several authors have discussed the basis of the relationships between the concepts of scales, scopes, and levels. Ryan, in particular, has pointed out that the notion of abstract hierarchical levels has contributed to a misunderstanding of the cause-effect relationships within emergent behaviors, sometimes even leading to circular arguments [111]. Defining scope as the 'spatial'[88] and temporal boundary of a system, "A property is a novel emergent property [if and only if] it is present in a macrostate but it is not present in any microstate, where the microstates differ from the macrostate only in scope." He further explains, "This class of emergent property arises from structure that is extended over the scope of the system… There is a difference between the local and global structure in any system that exhibits emergent novelty. This explains why emergent novelty cannot be understood or predicted by an observer whose scope is limited to only one component of a system." With that argument in mind, a behavioral hierarchy with abstract levels would be something overlaid onto some identified emergent behavior after the fact. Whatever one chooses to label as macro/microstate can only be rigorously justified after the appropriate scopes have been identified somehow. Ryan also defines and discusses the terms resolution, and scale. Unlike the sense commonly

[88] He seems to argue such that any metric space is suitable for scope/boundary definition, thereby generalizing beyond the length/time scales common in physics. Recall that 'the energy cascade' in fluid flow turbulence has scales in wave number space. Therefore, although it is uncommon to generalize as Ryan does, this concept is not totally foreign to physics. Also, the 'boundary' of a system, in SE, has to do with determining the set of functions, interactions, and structural components uniquely, or most closely associated with the system in question. Thus, setting a boundary has as much to do with conceptually isolating a system from its environment as it does 'zooming in or out' (changing scope). The difference with emergent behaviors is that one stops zooming out once the behavior has been identified. That said, some causal phenomena appear to be scale-free. The interested reader is referred to Dr. Sapolsky's discussion of fractals [134].

employed by physicists, scale seems to refer to multiplicatively magnifying/diminishing the magnitude of some property so that it becomes easier to distinguish from other properties at some given resolution. Resolution, for the purposes of this thesis, is simply the observer's ability to distinguish between the behaviors of the components of a system, thereby contributing to the epistemological challenges of emergent behavior identification (but not ontological challenges). Febres associates scale, scope, and resolution with the length of the description required to communicate an observation of a process, and in so doing, argues against Ryan's definition of 'scale' [112]. Ryan's usage of scale falls outside the scope of this thesis, as do Febres' arguments.[89] However, there is a sense analogous to scale in which quantities associated with emergent properties, as will be defined in this thesis, can be multiplied (or more appropriately, exponentiated) and directly tied to their resolution, according to Ryan's use of the term (see Section 3.2).

There is a consensus in the literature that emergent behavior ought to refer to behavior that cannot be described by simply aggregating two or more independent system behaviors [113] [111] [114]. Szabo & Teo make this argument along semantic and set-theoretic lines [96]. In arguing that emergent properties do not refine (referring to a discontinuity of language compatible with Szabo & Teo's argument), Polak & Stepney essentially argue that emergent properties, and hence behaviors, cannot be decomposed [115]. In each case, using terminology unique to their discipline, "non-aggregative" behavior is simply non-SE terminology for behavior that cannot be decomposed using

[89] Febres' arguments and metrics rely heavily on the length of information description, which will not be used in this thesis in that same way. Note that Ford's thesis [26] also relies on length of information description, but does not extend to emergent behavior, remaining instead within the context of decomposable behavior. This thesis will use dimension in a manner analogous to length of information description.

standard techniques.[90] Philosopher of Science, Dr. Wimsatt, however, elaborates further on what makes a system property non-aggregative, outlining four requirements for *aggregativity* [116]: (1) Inter Substitution: the property remains invariant as the system's components are rearranged (2) Qualitative Similarity: the property must be qualitatively the same after the addition or removal of components (3) Decomposition and Reaggregation: the property remains invariant under decomposition and re-aggregation of components, (4) Linearity: there are no cooperative or inhibitory interactions between system components. Although these conditions appear to be readily compatible with decomposable systems, Wimsatt argues that the only truly non-emergent property in existence is mass.[91] Some authors resist the idea that the overwhelming majority of the universe is comprised of emergent objects (e.g. [117]). However, this author considers Wimsatt's argument perfectly reasonable (at a philosophical level) given that the majority of object types in this universe are not-quarks, the majority of organism types on Earth are not-single-celled, and the majority of mathematics is not-linear, and so on. Ryan also discusses the significance of non-aggregativity, saying, "a Gaussian distribution is not organized, nor is it an emergent property of [independent, identically distributed] components," concluding that "… nonlinearity is a necessary condition for emergent properties" [111] illustrating once again that most of the concepts underlying behavior hierarchy/heterarchy, functional decomposition, complex behavior, emergent behavior are

[90] Anticipating an objection: this applies exclusively to non-axiomatic properties/behaviors. Axiomatic properties cannot be decomposed by definition. The decision of what element/property to treat as axiomatic is outside the scope of this thesis, as will be explained in the subsequent chapters.

[91] To his knowledge, of course. There may be others that fit his criteria.

interchangeable (see Figure 6).[92,93] The intersection of this terminology represents the extent to which the usage of these terms in the SE, SoSE, and philosophical literature is relevant to this thesis. To eliminate confusion among the technical terms used in this work, the SE use of "complexity" can be subsumed into "emergence" and "complex behavior" into "emergent behavior" for the remainder of this thesis.[94]

One important note on the subject of aggregating behaviors is that although several authors repeat the phrase "the whole is greater than the sum of its parts," Kubík points out that many authors fail to define what they mean by "sum of the behaviors of individual parts" [114]. Szabo and Teo, as well as Kubík, treat the term "sum" as a union of behaviors in order to apply set theoretic operations to their grammar-based approach [96]. However, "sum" is most commonly associated with addition and linearity. Ryan provides one of the clearest explanations, stating that "superpositionality, averaging and other linear operations cannot be the *source* of emergent properties… because a linear operator evaluates equally for any arrangement of the components… so the global structure is always exactly the sum of its parts" (emphasis added) [111]. This does not mean an emergent object cannot have a linearly computed property. It simply means that some linear property of a set cannot be the source of emergence. For example, a block of ice can be said to have a center of gravity and its macroscopic dynamic behavior can be predicted using classical mechanics, but it is

---

[92] Anticipating another objection: this is the steady convergence of ideas, and not the trivial consequence of different authors gradually co-authoring work. See similar remarks by Phelan [172].

[93] This is meant in the general sense. If an engineer is handed two sets of properties, it may be an association fallacy to equate the non-aggregative properties with the non-decomposable properties. For example, consider a set of atomic properties and molecular properties. There may be a non-decomposable molecular property in that list, as well as a set of non-aggregative atomic properties. The mere fact that they possess these attributes is not enough to prove that the former is caused by the latter.

[94] This is a statement on terminology and scope of research, not a formal definition of either term. A definition will be proposed in CHAPTER 4.

the interatomic forces that generate the block of ice, not the mere averaging of atomic positions.

Despite the challenges in emergent behavior decomposition, Kubík argues "there is no reason (at least at present) to think that there are phenomena not reducible to micro-macro relationships" [114]. This thesis agrees, so long as "reducible" is not construed to mean decomposable in the traditional sense. Thus, a gap in the SE, physics, and philosophical literature is:

<u>Gap:</u> There is no method by which component interactions can be used to predict the existence of an emergent system-level property or behavior and *traceably* attribute a quantifiable, system-level property or behavior to that system.[95]

Here, the existence of a system is contingent on the interaction of its components, not merely the collection of its components (as suggested by the model-based definition). This gap leads to the overarching objective of this research.

<u>Research Objective:</u> To develop a method for rendering non-decomposable, quantifiable SoS properties and behaviors traceable to the patterns of interaction of their constitutive systems, so that exploitable patterns identified during the early stages of design can be accounted for.

In order to achieve this objective in a manner conforming to standard engineering practices, a mathematical approach is needed for identifying non-decomposable behaviors,

[95] In the time since this thesis was proposed, two studies were found that outlined methods for associating emergent behaviors with a system's components (to some extent): [352] [353]. Discussions of these studies have been added throughout the thesis. The study in [208] was known prior to the proposal, and is discussed in CHAPTER 3.

identifying components engaged in relevant patterns of interactions, and tracing the aforementioned behaviors to those components.

<u>Gap</u>: There currently exists no single mathematical method that performs all the steps needed to satisfy the research objective.

While a variety of tools and techniques exist, the research presented thus far indicates that the primary reason no such method has been developed yet is due to the scientific preference for reductionist explanations of phenomena, built on reductionist ontologies.

<u>Gap</u>: An ontology that accommodates emergent behavior and enables falsifiable claims of system "existence" is needed as a philosophical foundation for a mathematical method.

It is also clear from the broader literature that there is substantial disagreement over the definition of terms such as "complex behavior" and "emergent behavior." This makes it harder to develop an ontology since it confuses the discussion of various topics.

<u>Gap</u>: Some acceptable baseline set of definitions for emergent behavior is needed in order to build a useful ontology.

These gaps lead to the first question that this thesis must answer:

<u>Research Question 1</u>: Which essential features of emergent behavior constitute necessary conditions that can be implemented in a mathematical/computational model?

So many definitions and types of emergence have been proposed and debated over its 2,300 years as a concept in Western literature[96] that Abbott has recently called for avoiding the term altogether saying, "given its burden of intellectual baggage there is little reason to continue to use it" [62].[97] Philosopher of Science Laurent Jodoin remarks on the sheer number of definitions and volume of literature, saying that it is easy to get lost or inadvertently confined within an incomplete or biased framework [118].[98] The conceptual stance taken by this thesis will generally align with the ideas, criticisms, and analyses of Mitchell, Baas, Ryan, Kubík, Wimsatt, Crutchfield, Abbott, and Minati. This work distinguishes itself from those authors largely via its methodology and hypotheses. Readers interested in summaries and reviews of various types of emergence are referred to [114] [97] [118] [59] [119]. Of the categories frequently discussed in the literature, the term closest to a behavior that is not decomposable but is traceable would be ***Weak Emergence***,[99] which is defined as any system-level behavior generated by component-level behaviors that can only be observed by simulation (or direct empirical observation) [120] [121]. In other words, there is no obvious way to predict weak emergence simply by knowing the rules that govern the behavior of a system's parts, just as one cannot predict the center of gravity of a water molecule simply by knowing the properties of hydrogen and oxygen.

[96] Many papers on Emergence trace the concept back to Aristotle [72], and draw a line through Western literature [58]. Readers interested in how complexity/emergence played a role in Eastern thought are encouraged to read "Chinese Medicine and Complex Systems Dynamics" [78].

[97] Readers interested in a review of the term's history are referred to Appendix A in Kim's thesis [58], and [283]. Standford Professor and Neuroendocrinologist Dr. Robert Saposlky has also published a relatable and very insightful series of lectures/talks on Reductionism, Complexity and Emergence [280] [134] [281]. Neither researcher discusses Post-Modernism or Reductive Analysis despite their overlap [73].

[98] His thesis utilized 633 references to analyze the relationship between emergence and entropy.

[99] See also an interesting discussion of strong and weak emergence by Lawhead [396]. Not only do the concepts in his paper somewhat resemble this thesis, his exploration of constraints to dynamical systems can be extended to the work done here.

One noteworthy study by Kokar, Singh, and Lu used Formal Concept Analysis [122] on a subset of the literature to identify eight concepts often associated with emergence [97]. The listed concepts not already explicitly discussed include: radical novelty, unpredictability, irreducibility, dynamical, coherence, decentralized. Radical novelty is described in terms of the organization of components such as "new structures, patterns of behavior of properties" [97]. The authors appear to at least partially conflate unpredictability with irreducibility. When some writers refer to unpredictability, they are generally referring to the epistemological challenges of emergence identification (including arguments against treating the subjective experience of surprise as a necessary condition) [123] [101] [124]. This thesis will focus more on ontological questions of emergence identification, and strive to expect the unexpected. Irreducibility is presented in a manner synonymous with non-decomposability, wherein an emergent property "is irreducible if [it] cannot be deduced from the properties of its constituent parts" [97]. Dynamical means that emergence is the result of changes over time, and so the models used here will include time. Decentralized means that no single entity is controlling the behavior of all components. Unfortunately, in cases where every entity follows the same rule set (e.g. atoms obeying laws of physics), or every entity follows a different rule set (e.g. toddlers doing anything at all) this concept does not provide a mechanism for distinguishing emergent from non-emergent behavior. Cases between these extremes are an interesting area of research, but systematically studying these possibilities falls outside the scope of this research. Finally, coherence is simply referred to as "logical consistency or quantitative continuation" [97]. On the surface, this appears to be a departure from the statistical correlation sense of the term used by Goldstein (their reference), "emergents appear as integrated wholes that tend to maintain some sense of identity over time. This coherence spans and correlates the separate lower-level components into a higher-level unity" [119]. Although Goldstein's reference to correlation is inspiring (see Sections 3.1-3.2), neither work develops the notion of coherence further.

Supervenience can be considered another necessary condition of emergence [118], but has become something of a problematic term. Although it remains popular in the literature on emergence, its use in the larger philosophical literature has changed over time [125]. Philosophers often discuss whether a property ought to be classified as emergent, supervenient, or causal[100] using various nuanced definitions [100] [126] [127] [128] [129] [130].[101] Its original use is largely equivalent to the current use of emergent behavior if that behavior is attributed to a SoS: a property is supervenient if it exists in addition to the properties of the constitutive systems that generate it [125]. In this sense of the term, using supervenience as a N.C. would create a circular argument.

Grammar-based methods for emergence identification, most of which have a set-theoretic component to them, are only suitable for use after emergence has been identified and named by a subject-matter expert (Szabo & Teo refer to this as "post-mortem" identification [131]) [114] [132] [111]. They are post-mortem because the determination must first be made that something is emergent before it can be named and assigned to a set. Implementing these approaches would amount to reverse-engineering or validating an emergent behavior [111] [131], unlike a N.C., which is an *a priori* criterion used to

[100] This ties directly back to the discussion on downward causation. To say that "the property of a SoS supervenes on the property of a system" does not necessarily imply it can affect that property (see references).
[101] See also discussion in Section 1.2.1 of [133].

facilitate prediction. Another method compatible with grammar-based approaches is the classification scheme by Chen and Clack, which borrows the scope and resolution definitions from Ryan and adds the qualification that “emergent properties are those not explicitly defined in the component specifications” [133]. In the context of their approach, this means that an emergent behavior can be characterized by tracking the change of some property over time, subject to a rule (a conditional statement that triggers the code to change the property). Although they do not go into details on specific implementations, they suggest that those rules are associated with the organization of the components being modeled. The behavior is called complex if the cause-effect relationships are unclear. While this is consistent with the previous discussion on complexity, this approach is also post-mortem and cannot be a necessary condition here.

Some authors use the changes to the length of the description of an event as indicators of emergence (see Kolmogorov complexity later in this section). Although these authors rarely refer to mathematical syntax (which can sometimes appear to use deceptively compact notation), it is reasonable to think that replacing one set of equations for low-level behavior with a suitable set of equations for high-level behavior could result in a change in description length.[102] Febres [112] also explores the notion of two-dimensional description lengths in a context tangential to emergent behavior. Although the work by Febres does not have direct application here, a notion of model description changes will be used in this thesis (see Sections 3.1–3.2 and CHAPTER 4).

[102] Since computers implement mathematical models, there are two direct analogies in computer science: (1) replacing one machine-code computer program with a shorter program, (2) replacing a high complexity algorithm with a low complexity algorithm.

As with complexity, a variety of quantifiable measures of emergence have been proposed. Before proceeding to metrics aiming to predict emergence, it is worth noting that various nonlinear behaviors have been associated with emergence and thus, in some sense, are considered to indicate it (these appear in the complexity literature as well). They include chaos, fractals, bifurcations, and Turing instabilities (see reviews in [134] [135] [59] [118]). Although bifurcation is a feature of certain classes of differential equations, physicists associate bifurcation with physical structures of atoms undergoing some form of symmetry breaking (see [110] [136], and the discussion on self-organization in [118]). Thus, again, the mere fact that components have one or more arrangements that persist over some time interval is a N.C.[103] Applying this to some scenario would require observing a change in arrangement over time, which, when performed spontaneously, is referred to as ***self-organization***.[87] Recall that these nonlinearities only indicate emergent behavior when they occur in the context of some form of interaction, as is the case with self-organization since there must be some underlying mechanism that generates and perpetuates the structure.

As with complexity, there are several information-theoretic measures of emergence. Some measures compute the so-called statistical complexity of the information in the model (see references in [133]), which can potentially indicate that emergence has occurred although it might be difficult to single out the emergence-causing interaction in a model depicting multiple simultaneously occurring interactions.[104] Fuentes defines the total information of a model [134] as the sum of a measure of system complexity and a measure

[103] Whether that structure is symmetric is case-specific, and thus not a N.C. considered here.
[104] Most statistics discard or mask sample-specific information by definition.

of system disorder. Fuentes calculates the system complexity using a function that takes the value of the control parameter of a differential equation (e.g. a coefficient from an equation characterizing bifurcation) and computes the Kolmogorov complexity, while the measure of system disorder is a function of its information entropy. The ***Kolmogorov complexity*** is defined as the length of the shortest description of an object in binary (e.g. the number "13" has the 4-digit description "1101") [137]. However, the Kolmogorov complexity is generally not a computable function, nor can it be approximated in a practical way, making Fuentes' metric intractable [137] [138]. Nevertheless, the notion of measuring the complexity of a model (loosely speaking, similar to Fuentes) is not often found in the literature and will be used in this thesis. Information entropy measures, such as the Shannon information entropy, are often associated with emergence [66] [118]. Since information entropy has an analogy with thermodynamic entropy,[105] and thus energy, authors have also considered computing the energy of system components in order to identify emergence. However, such energy and entropy metrics discard substantial amounts of information (energy in particular discards structural information [80], and entropy is not one-to-one),[106] making them post-mortem techniques.

Still other metrics have been considered. Chan proposes measures based on counting the number of interactions (the measures themselves are statistics) occurring within an Agent Based Model [139].[107] Although it is interesting to observe the relationship

---

[105] As stated earlier, some citations use the term "complexity" rather than emergence. See Section 1.5 for additional references computing some form of entropy.
[106] For example, the Shannon entropy is closely related to the *expected value* of the Kolmogorov complexity, meaning that behaviors (via their signals) are analyzed collectively, not individually. Thus, as the number of system component interactions increase (each potentially increasing or decreasing entropy), the ability to single out the relevant emergent behavior diminishes.
[107] Still more methods applied to (social) networks are listed in Chan's references [138].

between interaction distribution shapes and the patterns of interaction they correspond to, Chan does not argue that this process can be reversed (that knowing a distribution implies knowing the pattern that will take place). Rather, Chan observes that the distributions deviate from normal when emergent behavior is present, and appear normally distributed in the absence of emergent behavior. Fisch et al. use measures of divergence [140], which detect a shift in the distribution of information entropy as the behaviors of a set of systems changes over time, and then measure the dissimilarity between the two distributions. These metrics are later adopted by Kim to perform design space exploration [59]. Kim argues that one can avoid or encourage the generation of emergent behavior by identifying regions of the design space that contain sharp changes in divergence measures. Kim's approach, however, does not go so far as to say what those emergent behaviors will be. Hovda measures the "amount of simulation" required to generate weak emergence [141]. However, Hovda's method identifies propositions about the system, rather than emergent properties of the system. Finally, Seth proposes using a nonlinear adaptation of Granger causality to identify weak emergence (he refers to this as G-emergence) [142]. Granger causality is named after the mathematician that first introduced a linear regression technique for statistically determining whether a variable X, for example, has a larger influence on the time-evolution of variable Y, than, past values of Y itself. As proposed by Seth, this technique will be revisited in Section 5.1.1 as a means for identifying causal relationships between low-level and high-level properties. However, the calculation of G-emergence, itself, is only meaningful after some property has been determined to be emergent. Since (in the abstract) the number of higher-level properties is potentially infinite, there remains a methodological gap in the literature, which this thesis will strive to fill:

<u>Gap</u>: There is no method in the literature for determining the number of emergent properties a system can have.

To the extent that complexity and emergence are interchangeable, the aforementioned argument against complexity measures (which have the potential to conflate complexity with complicatedness) can be used against these measures of emergence. Like the grammar-based approaches, many of these metrics can only be applied post-mortem. It does not appear that any metric has gained wide acceptance (some are too recent to have gained traction). Although a thorough comparative analysis of these metrics on some canonical case would be an excellent contribution to the literature, it is outside the scope of this thesis. This is due, in part, to the fact that it is unclear just how many of these metrics could be applied to the same canonical test case. For example, it is unclear that the transient length, clustering coefficient, or exponential growth proposed by Dogaru [143] will extend beyond cellular automata in an unambiguous manner.[108] For example, a clustering coefficient is not one-to-one for clusters of objects that have different shapes (a collection of squares versus circles), and that becomes worse when those shapes are allowed to vary in size. Furthermore, to be considered canonical, the test case would require a consensus on the emergent behavior it exhibits. Such consensus only appears in the literature for the simplest cases. For example, there is *some* qualitative consensus on the

[108] Of the three, the transient length is the easiest to generalize, but not only is the equilibrium condition it depends on case-specific, it may not be unique in other applications.

emergent behaviors demonstrated in the Boids model, but little agreement on what quantities to associate with that emergence (see Section 3.3).

The collection of candidate conditions reviewed thus far is insufficient for application to SoS modeling as required by the Research Objective. For example, there is no mechanism/justification for making the ontological leap from a collection of quantities to the declaration of the existence of a thing.[109] Then, there is no method for determining the number of properties the SoS has, which translates to an inability to predict/model its behaviors or interactions. Finally, there exists no decision procedure (or even analysis procedure) for selecting one or more properties to assign to the SoS (these gaps will be formally restated in CHAPTER 4). To restate the problem in layman terms,

> *Any six-year-old can give you a dozen examples of physical objects, and most people with at least an undergraduate course in philosophy can also give examples of non-physical objects. But if asked to produce a definition of 'physical object' that adequately captures the distinction between the physical and the non-physical, the average person can offer little more than hand-waving.* [51]

A point perhaps missed by many reductionists is that a collection of interacting atoms forming a molecule (or at least our description of it) is just as abstract as a collection of interacting quarks forming a proton, which is just as abstract as a fleet of ships, or continuous materials. In this sense, the problem in the emergence literature is the same problem faced by systems engineers, or physicists, or ship designers performing fleet synthesis studies. It is relatively easy to informally identify an object, but it can be very challenging to rigorously define and model that object.

[109] This statement is deliberately broad since, again, the model-based definition permits a SoS (the "thing") to be anything from a collection of atoms to a collection of military ships.

One of the more influential objections to the concept of emergent behavior was written by Philosopher Dr. Jaegwon Kim. An accessible review of his arguments, particularly his causal exclusion argument, is presented in [126]. Essentially, Dr. J. Kim's causal exclusion argument states that any explanation of physical cause/effect attributed to an emergent property is redundant (and therefore unnecessary) because it implies the existence of another explanation in terms of the lower-level components engaged in the complex behavior (the lower-level components presumably being those that are truly real). While this is perhaps compelling at a philosophical level, engineers take for granted that real pipes are actually built to transport real water.[110] With this perspective in mind, Philosopher of Science Dr. Mitchell writes, "The standard philosophical notion of emergence posits the wrong dichotomies, confuses compositional physicalism with explanatory physicalism, and is unable to represent the types of dynamic processes… that both generate emergent properties and express downward causation" [144]. Referring to Dr. J. Kim's arguments against explanations of the natural world in terms of emergent behavior [145], Mitchell criticizes the assumptions on which his argument is founded (referring to one of humanity's, and thus science's, glaring limitations),

> *All descriptions are abstractions or idealizations. They do not stand in a one-to-one mapping relationship with the entirety of the undescribed world. To think that our language [including mathematics] captures the physical world exactly is simply misconceived. Descriptions are always partial… If there is something in the world that can be isolated by the functional description (caused by X and causing Y), there is no reason to think that a physical description of that piece of the world, partial as it is, will be identical with a higher-level description of that piece of the world, partial as it is…* [144]

[110] As opposed to designing systems of quarks to move systems of quarks.

In other words, there are multiple, partial, equally valid ways of representing phenomena from different levels of abstraction, each capable of describing some aspect of nature in a manner not possible at the alternate level. It is unreasonable to conclude that the behavior of a collection can always be explained in terms of the behaviors of it parts. More counterarguments to Dr. J. Kim can be found in [146].

In an effort to move past the apparent paradox of trying to decompose the non-decomposable, CHAPTER 2 - CHAPTER 3 will develop additional, supporting theory and present some hitherto unstated definitions. CHAPTER 4 will present the research questions required to address these gaps, as well as the hypotheses that attempt to answer those research questions.

CHAPTER 2. MODELING CHALLENGES AND DEFINITIONS

By definition, a Search and Rescue mission is not an emergent behavior. It is immediately decomposable into two sequential behaviors. However, SAR missions involve using a Directed SoS, and often operate within a Collaborative SoS, both of which can exhibit emergent behaviors not directly associated with the SAR, but which have an impact on their performance. SAR missions are a natural fit for a Fleet CBA as well as a Fleet Synthesis Study. To the acquisition community conducting a CBA, this raises two immediate questions: (1) what are the emergent behaviors exhibited during the mission, and (2) how can the impact of these behaviors on mission performance be measured? A third question would naturally be, how can the emergent behavior be exploited? The answer to the first question will follow from the discussion in CHAPTER 4. The second and third question are the subject of this chapter. Since the subject of this thesis is emergent behavior, however, this thesis will rely on a notional combat model rather than a SAR model. Notional combat models have the advantage of exhibiting self-organization since all combat involves self-organization, and there exist many well-documented battles in history to compare to. The combat model used here will be based on WWII-style dogfighting, and will be discussed in Section 2.4.

2.1 Measures of Merit

The acquisition community has a now well-established practice of measuring ship performance using Measures of Effectiveness [10], or, more generally, Measures of Merit. Borrowing the taxonomy synthesized by Hootman and Whitcomb, a ***Measure of Merit*** (MOM) is any metric "that characterizes a system under analysis" [147]. MOM's are hierarchically arranged from the specific design parameters that characterize the physical structure and capabilities of a weapon system or ship (i.e. component and system level), up to the metrics that measure the performance of a fleet (SoS level), as depicted in Figure 9.

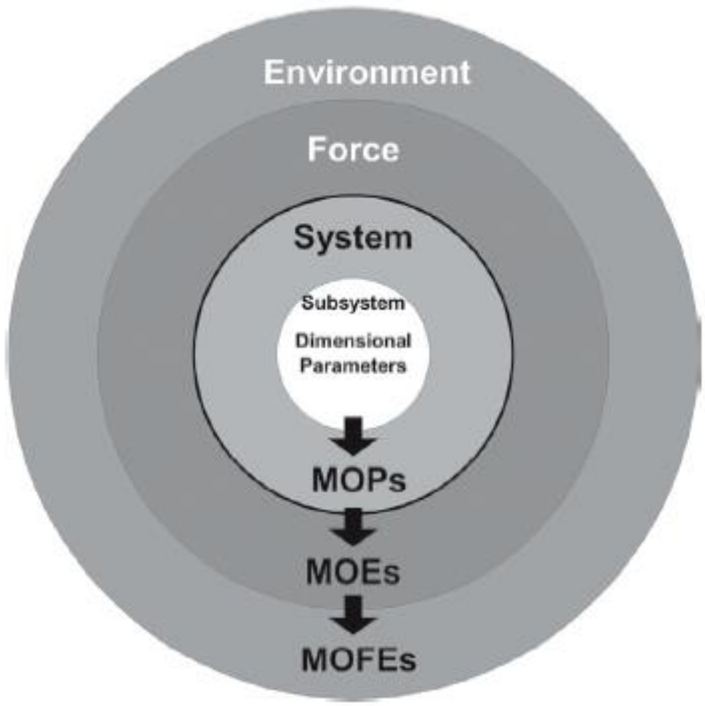


Figure 9 –Measure of Merit Hierarchy, reproduced from [147]

A ***Dimensional Parameter*** (DP)[111] is usually a physical subsystem specification (e.g. radar aperture size), while a ***Measure of Performance*** (MOP) represents subsystem performance (e.g. radar gain), and a ***Measure of Effectiveness*** (MOE) corresponds to the system's ability to execute a task (e.g. probability of detection) [147].[112] A ***Measure of Force Effectiveness*** (MOFE) would correspond to the SoS[113] ability to execute a certain task (e.g. search). Within a level of abstraction, each metric ought to be as independent as possible.[114]

---

[111] Some engineering design parameters can be non-dimensional. To avoid confusion, the reader can substitute design parameter for dimensional parameter without loss of generality.
[112] All examples listed in this sentence are reproduced from Table 1 of [150].
[113] Hootman and Whitcomb use the term "supersystem" to refer to a SoS [150]. They suggest that the ship as well as the strike group the ship operates in be treated as the SoS. The radar system appears to be considered system-level in their work.
[114] That is, decoupled and statistically independent where applicable.

Once the relevant MOMs have been enumerated, the goal, according to Hootman and Whitcomb, is to combine these MOMs for design and acquisition using a set of models that can calculate the various levels of performance relevant to a designer or decision-maker (including the fleet-level), as in the framework developed by Koleser [148].[115] Hootman and Whitcomb argue that a ship's performance *cannot* be properly understood in isolation, writing "we must look beyond the total ship system to the battleforce; engineers must consider how the system that they are designing interacts with the environment it operates in and the other systems it operates with" [147]. Not only is the SoS performance important for understanding ship performance, but the mission itself also plays a central role.

The next step to determining what impact an emergent behavior may have on mission performance is to develop a procedure for selecting adequate MOMs. Hootman and Whitcomb refer to the *goal-question-metric* (GQM) method.[116] As the name suggests, this method begins with: (1) listing the goals to be achieved by a system, which in this case are equivalent to the goals of the mission, (2) reframing the mission objectives as questions that characterize the manner in which the objectives will be achieved (while simultaneously ensuring the accompanying mission model can quantitatively represent that characterization [149]), and finally (3) identifying the quantitative metrics that answer the aforementioned questions. Hootman and Whitcomb use the mission of a submarine as an example, raising the question "What is the probability of the [submarine] avoiding detection?" to which the answer is a metric that computes that probability [147]. Another approach for MOE generation was proposed independently by Sproles [150],

[115] In practice, large collections of interdependent computational models can require large amounts of computing power to execute. Surrogate models and response surface techniques are used to accelerate this process [150]. Although these practices are ubiquitous in modern computer-aided design, the details of this practice are largely outside the scope of this thesis.
[116] Hootman and Whitcomb attribute this method to Kowalski et al. but provide no citation. It appears to originate in a 1994 paper by Basili, Caldiera, and Rombach [152], based on Basili's earlier work [286].

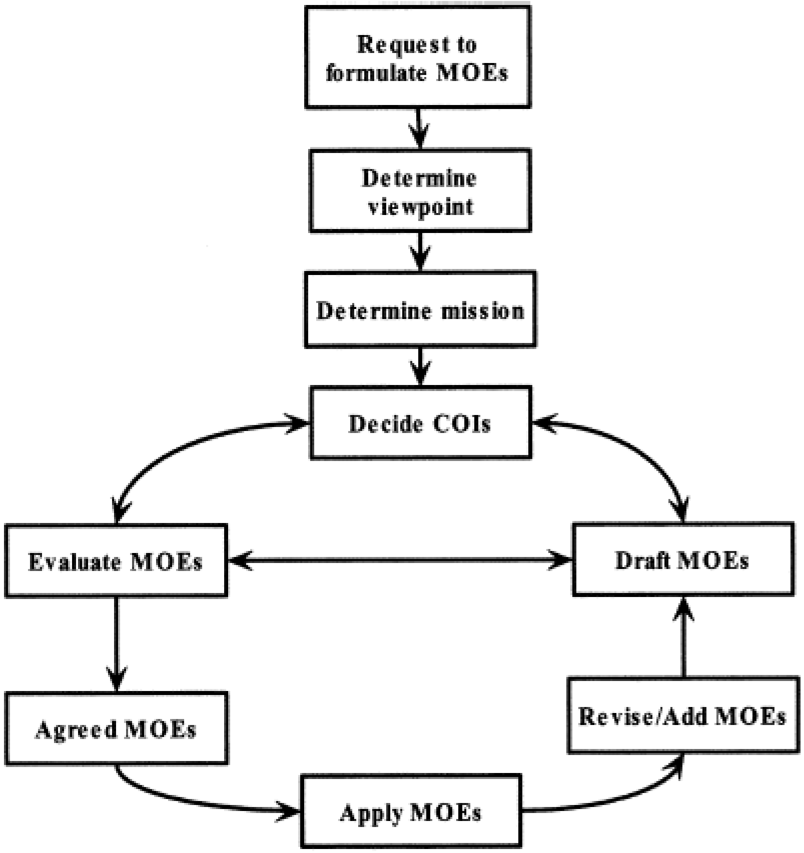


Figure 10 – MOE development method by Sproles, reproduced from [150]

The method by Sproles defers substantially from GQM. First, the initiation of MOE development precedes the mission definition. To Sproles, a MOE specifically answers the question “Does this meet my need?” [150] Second, the priorities of the stakeholder must be taken into consideration (the viewpoint step) before mission selection. He illustrates this with an example from WWII where the Allies could have considered “the number of U-boats sunk” as a priority, versus “the number of merchant ships saved.” In this case, the two priorities guided the design solution in either an offensive or defensive direction. From a CBA standpoint, this difference would impact the portfolio of missions considered relevant to the acquisition program, which can result in wildly different designs. One would then expect that with different designs come different (emergent) behaviors. In Section 3.2 of her PhD thesis, Dr. Griendling reviews a third metric derivation approach [56], the

Practical Systems/Software Measurement (PSM) method (see also [151]), and incorporates it into the Relational-Oriented Systems Engineering Technology Tradeoff Analysis (ROSETTA) method for metrics derivation. However, her approach is intended for capabilities and behaviors that are decomposable, which does not apply here. Furthermore, both GQM and PSM rely on the idea that a particular property or behavior is directly measurable. Although taking measurements is fundamental to any experiment, Section 1.7 discusses why direct emergence measurement techniques also fall outside the scope of this thesis. Thus, GQM and PSM could only be applied indirectly, if at all. Finally, for the purposes of this thesis, the mission will be taken as given, acknowledging the risk that one mission type, with one parameter set, may exhibit many more emergent behaviors than another, which will either facilitate or frustrate this effort.

The remaining step for determining what impact an emergent behavior may have on mission performance is to select a model with which to compute the relevant metrics. Here, however, SoS exhibiting emergent behaviors are faced with two mutually reinforcing problems. First, since measures associated with complex behavior tend to be strongly nonlinear, they do not necessarily correlate with the success of a mission in the way a straightforward application of GQM or PSM might suggest. In real applications this can mislead decision-makers, thereby contributing to tragic losses of life, as occurred during the Vietnam War [152]. Second, as research by RAND Corporation pointed out, combat models can be very inaccurate in part due to incorrect assumptions on the part of the developer [153]. Although it may be tempting to think that SAR mission models will somehow fare better, the problem of unrealistic models is actually universal.

2.2 The Limitations of Modeling & Simulation

As with acquisition, SE, SoSE, complex systems, and emergence, modeling and simulation is a very broad field that extends into multiple disciplines. Citing that "no single, strict and widely accepted taxonomy of modeling techniques exists," [81] Balestrini-Robinson devotes Section 2.3 of his PhD Thesis to thoroughly reviewing a number of modeling taxonomies, their various subcategories, and various practical concerns surrounding their use.[117] In order to maintain focus on emergent behavior detection, however, this thesis will take a largely philosophical approach to the discussion on modeling, and introduce only the categories that would be relevant to a computer programmer: a model, an algorithm, and a computer program.[118]

Sayama informally defines a ***model*** as "a simplified representation of a system. It can be conceptual, verbal, diagrammatic, physical, or formal (mathematical)" [154]. For the purposes of this thesis, the term will be reserved for mathematical models only (i.e. systems of mathematical equations that characterize the properties and behaviors of a system)[119]. Sayama also defines a ***dynamical system***, again informally, as "a system whose state is uniquely specified by a set of variables and whose behavior is described by predefined rules" [154]. To avoid confusion, recall that the lowest-level objects considered in a model are the components. For reasons that will be explained in Section 2.2.2, the components of a system being represented by the model can be thought of as dynamical systems, while the higher-level objects (system, and SoS) may exhibit emergent behaviors for which no rules have been written. Suppose, now, that there existed a data set (measurements of some behavior) that a scientist wished to characterize using a model.[120] Richardson writes,[121]

---

[117] Section 2.3.2.1 of his discussion focuses the DoD taxonomy, MOMs, the models that calculate them, and the acquisition stage during which those models are utilized.
[118] Going forward, a computer program will be simply referred as a program or simulation.
[119] Properties are often referred to as ***state variables*** in the literature (for example, page 30 in [157]).
[120] This extends to the laws of physics themselves.
[121] This is a summary provided, in part, for narrative purposes; not as an appeal to authority. See Section 1.5-1.7, and CHAPTER 4 for the detailed argument.

"[There] are an enormous number of qualitatively different ways to model the same phenomena… [Given] any amount of evidence, there are mutually incompatible models which equally fit with the evidence … [and] when a prediction from a model contradicts the observation, there are various mutually incompatible ways for making the model compatible with the evidence… [A result] is that even if our models can be used to develop causal explanations… we cannot be sure that those explanations bear any relationship to reality whatsoever." [122] [155]

Even at the level of physics, this is an inescapable limitation brought on by complexity. Again, Laughlin writes regarding behaviors "governed by emergent rules… in practice, if you are locked in a room with [a low-level model], you can't figure the rules out in the absence of experiment, and hand-shaking between theory and experiment" [75]. Military mission modeling is only made worse by the inclusion of human decision-making, for which there are no widely accepted "laws" as there are in physics. Thus, the skepticism surrounding combat models should not be surprising. Richardson quotes mathematician John Maynard Smith, who said,

"[I have a] general feeling of unease when contemplating complex systems dynamics. Its devotees are practicing fact-free science. A fact for them is, at best, the outcome of a computer simulation; it is rarely a fact about the world." [155]

For this reason that the SE Handbook warns, "The systems engineer must continually distinguish between systems in the real world and system representations" [53].

An experimentalist might ask, "Why not rely on experimentation instead of modeling?" Clearly, experimentation in a military context is subject to severe moral, societal, financial, and practical constraints. Setting the obvious aside, complexity spares no one. Thus, the epistemological and ontological counter-argument to experimentation is: instability. Discussing the far-reaching effects of static, dynamic, and structural

[122] There is always hope.

instabilities, Schmidt notes that complexity creates challenges for the four methodological prerequisites of modern science, “reproducibility / repeatability, predictability, testability, and describability / explainability” [156]. Regarding reproducibility Schmidt writes, “Instabilities convey unobservable small effects to the empirically accessibly scales… and by this, instabilities induce problems regarding experimentation… the lack of control is not just a pragmatic or epistemic boundary that could be overcome by improvement of methods and more advanced technology… it is inherent in physical objects.” [156]. Regarding predictability Schmidt then goes on to write how approximations as basic as truncating real numbers (since computers cannot store infinitely many digits) can make predicting certain phenomena impossible due to instabilities that are extremely sensitive to small perturbations. Prediction is also limited by the fact that computers are physical machines governed by thermodynamic laws (such as the generation of entropy) that place a limit on what can be computed. Thus, these limitations are objective. Regarding testability, Schmidt writes, “for any unstable model that refers to an unstable object ‘details of the dynamics, which do not persist in perturbations, may not correspond to testable […] properties.’” [156] In other words, the phenomenon must be stable enough to be repeated during an experiment, or it will be impossible to test. Finally, regarding explainability, Schmidt writes “Unstable processes cannot be reductively condensed into a simple law… According to von Neumann’s idea on complexity, a complex process is defined as one for which the simplest model is the process itself. The only way to determine the future of the system is to run it” [156].[123] These are reiterations of the reasons why emergent behavior cannot be predicted from the properties and rules of component systems. Emergent behavior can only be observed once those components are placed into an environment (physical or digital, hopefully with some compatibility between their results) and permitted to interact over some time period. Unfortunately, the problem with experimentation is that the scientist has

[123] This is the unpredictability referred to by Kokar et al. [97]

little to no control over the environment nor control over the most fundamental system components. The more constraint and isolation one imposes on a physical system, the fewer emergent behaviors can occur.[124]

Having removed any potential for naïve idealism, it is now necessary to impose one of two key assumptions required for this work:

Assumption 1: The model is valid.

This assumption means that model accurately represents every behavior the lower level components could possibly exhibit. Alternatively, this assumption means that the only behaviors of interest will be those exhibited by the low level components and collections thereof as permitted by the model. The issue of how to select the lower level rules in order to observe a desired behavior in reality (i.e. issues of selecting relevant information, and compatibility with experiment) are outside the scope of this thesis.[125]

Of course, it is not enough to suppose that an assumption will enable meaningful research. Given the challenges to modeling that emergence and instabilities present, Schmidt provide suggestions. The first is to avoid the error made by scientists in the past: "In order to counteract the problems raised and to reject the methodological crisis, the very first attempt is always to re-introduce dogmatically what seems to have been lost" [156]. Although Schmidt is referring to instabilities, one might argue this thesis has already committed this error at least once in Assumption 1. However, Assumption 1 is justified in the sense that it is a necessary condition. If the concepts and mathematics presented here

[124] The interested reader is referred to Elif Shafak's tangentially related discussion of circles [298].
[125] Readers interested in methods for validating observations of emergent behavior are referred to Patrick Meyer's thesis [303], or the methods by Szabo, Teo, and Birdsey [131] [276].

cannot work in the most ideal case, they certainly will not work on experimental data except by sheer misfortune. Returning to instabilities, Schmidt encourages the implementation of unstable models in computer simulations. Rather than imposing stability on the mathematical equations, as past scientists have, one can simulate the equations and look for qualitative features that persist over time. “The qualitative still remains mathematical, but with a different meaning… These qualitative characteristics refer to the appearance of the phenomenon and geometry of the pattern *after* the process of time evolution – and not solely in the bare equation…” (i.e. emergent patterns / objects can appear while executing a simulation of equations that demonstrate instability) adding that “… Persistence is a necessary requirement for any empirical test and for physical relevance” [156].

Like Schmidt’s work, Richardson’s paper is intended to encourage a culture of thinking in terms of nonlinear modeling. Although it is not a paper of guidelines for building nonlinear models of complex behaviors, per se, he cites work by Allen regarding two fundamental modeling assumptions one can make, which are, (a) “no macroscopic adaptation allowed,” and (b) “no microscopic adaptation allowed” [155]. Relaxing assumption (a) leads to a model that can exhibit self-organization, while relaxing both leads to a fully evolutionary model. This thesis will only relax the first of Allen’s assumptions in a search for persistent patterns (see discussion on self-organization in Section 3.2).

An ***algorithm*** can be defined as “a specific set of instructions for carrying out a procedure or solving a problem, usually with the requirement that the procedure terminate at some point” [157]. Here, the algorithms of interest are the algorithms designed to calculate the solution to a model. For the behaviors of dynamical systems (particularly those exhibiting complex or unstable behavior), this means obtaining the values of each component's properties and behaviors over some pre-determined period of time (usually divided into a finite number of time steps). In other words, the algorithms in consideration here are approximations of the continuous solution to a set of mathematical equations (possibly differential equations) that cannot be solved analytically. A ***program*** can be defined as "a plan of action that is to be executed by an executor, usually an automatic device, most often a computer; instructions for an algorithm" [158]. For the purposes of this thesis, a program is an algorithm implemented on a computer. Since measuring the impact of an emergent behavior requires executing a computer program, the computational resources required to execute the program and availability of computational power place a practical constraint on how much data can be gathered, thereby limiting the volume and/or accuracy of measurements that can be taken.[126]

The mathematical representation of the relationship between the properties and behaviors of a set of systems, whether using a model, algorithm, or program, requires identifying a set of variables that store the data (the literal values of the property or behavior at a given moment in the simulation), and performing a series of operations in order to calculate the value of each variable at each moment in time. It is common to measure the difficulty of obtaining a solution to a mathematical problem in terms of the ***complexity*** of the algorithm or program computing the solution. Therefore, complexity can now be defined in two ways. First, the ***time complexity*** "of an algorithm A is defined to be the number *f(n)* of atomic instructions or operations that must be executed when applying A to any input set of measure n [159]".[127] An ***atomic operation*** is defined as an operation that "has a fixed constant number of operands" [159]. For example, adding two real numbers is an atomic operation. Adding two matrices of real numbers, however, has a complexity

[126] "Accuracy" can be used here because of Assumption 1.
[127] Leiss points out that, technically, this is the "worst-case" time complexity.

that depends on the size of the matrices, and thus is not atomic [159].[128] Second, the space complexity "of algorithm A is the amount of space, again as a function of the measure of the input set, that A requires to carry out its computations, over and above the space that is needed to store the given input (and possibly output…)" [159].[129] Since an algorithm can arrange any number of operations is more or less efficient ways, algorithm complexity can often be stochastic rather than deterministic (this will be revisited in CHAPTER 4-CHAPTER 5). The key distinction between algorithm complexity and typical measures of performance used in engineering is that complexity is based on the asymptotic behavior of the algorithm. That is, complexity measures the growth in the resources required to complete the algorithm as the number of input parameters, *n*, increases. This enables a machine-independent assessment of algorithm efficiency that captures the essential difficulty of the mathematical problem. The complexity of programs, on the other hand, is not only machine dependent, but often also compiler optimization dependent (never mind issues due to bugs, exception handling, rounding errors, passing parameters, etc. as discussed in Chapter 4 of [159]).

A subject this author has not seen covered in the literature, however, is a definition of ***model complexity*** that is analogous to the algorithm and program complexities mentioned here. This is probably due, in large part, to the relative infrequency with which models are called into question. For example, it is taken for granted that "F = ma." Engineering is largely focused on applying it. Mathematics is largely focused on solving it (given initial conditions, etc.). In general, however, it is tacitly assumed that the model "is what it is." Of course, it is true that one differential equation can characterize an enormous diversity of behavior (e.g. bifurcation). It is also true that one differential equation can

[128] Unless otherwise stated, this thesis uses word complexity, not bit complexity. A "word" is taken to refer to a real number.

[129] Programs (the implementation s of algorithms) can have a complexity different from the algorithm it is implementing due to a variety of reasons including: compiler optimizations, machine performance constraints (memory limitations, processor cooling limitations), choice of programming language, etc.

characterize all behavior relevant to some level of abstraction (e.g. no one is suggesting the Navier-Stokes Equations be discarded). However, if emergent behaviors can be qualitatively distinguished from other behaviors using a set of quantitative data, this implies there exists some way of qualitatively distinguishing one region of the solution to a differential equation from another. A qualitative distinction between two regions of the same data set can be represented using two different functions. For simulation purposes, this could take the form of some system components exhibiting stable behavior for one time interval, transitioning into unstable behavior, then re-stabilizing. In each case, a different function can be used to approximately represent the behavior (the domain of the function can be dictated by the desired accuracy of the regression). This is not new. In fact, it is quite common to simplify equations, obtain solutions to "special cases," and then stitch the solutions together (as in [160]). Referring back to the discussion in Section 1.7, if the space complexity of the model were to change, that would suggest either an increase or decrease in the amount of information required to describe a phenomenon, which, under the right circumstances, could indicate emergence. The thesis will argue that emergent behavior can be distinguished from other behavior based on stable patterns of interactions. Thus, let the ***space complexity of a model*** be defined as the number of dependent variables required to accurately represent the properties and behaviors of the components of a system (this permits the model complexity to scale with respect to the number of components). Let the ***time complexity of a model*** be defined as the number of operations required to express the relationship between the dependent and independent variables, scaled according to the type of operation. Calculating model time complexity requires much more nuance since one must be able to unambiguously determine the complexity of operations such as addition and multiplication, as well as non-elementary operations such as differentiation, or the evaluation of transcendental functions (see CHAPTER 4-CHAPTER 5, and Section 5.1.3 in particular). For calculations of model complexity to be meaningful, a second assumption is required. This leads to a second necessary assumption:

Assumption 2: The model is efficient.

To say the model is efficient means that there are no missing, excess, redundant, or otherwise unnecessary variables or constants.[130] In other words, the equations have been simplified without sacrificing solutions or singularities (provided singularities ought to be present). Assumption 2 should be interpreted as a combination of Assumption 1 and Ockham's razor. Due to the current ambiguity in calculating the time complexity of a model, it is impossible to guarantee that a model has minimum time complexity in the naïve sense. A stronger argument will be attempted in CHAPTER 4.

Having addressed issues of incompatibility between the model and reality, it is now possible to discuss the distinction between a system and SoS, along with the challenges of predicting SoS behavior, within the context of a program (a simulation).

2.2.1 Complexity-Appropriate Simulation and Tool Choices

The Complexity Primer (borrowing from Cook) provides a collection of tools and simulation techniques that are applicable to the study of emergence (see Figure 11). In order to avoid confusion with mathematical models, the term ***simulation*** will be used in place of model when referring to techniques such as Agent-Based Modeling.

[130] This is equivalent to saying that there is no coordinate system transformation that reduces the number of variables (e.g. writing the equation of a circle in polar coordinates rather than Cartesian).

ANALYZE	DIAGNOSE	MODEL	SYNTHESIZE
Data Mining	Algorithmic Complexity	Uncertainty Modeling	Design Structure Matrix
Splines	Monte Carlo Methods	Virtual Immersive Modeling	Architectural Frameworks
Fuzzy Logic	Thermodynamic Depth	Functional / Behavioral Models	Simulated Annealing
Neural Networks	Fractal Dimension	Feedback Control Models	Artificial Immune System
Classification & Regression Trees	Information Theory	Dissipative Systems	Particle Swarm Optimization
Kernel Machines	Statistical Complexity	Game Theory	Genetic Algorithms
Nonlinear Time Series Analysis	Graph Theory	Cellular Automata	Multi-Agent Systems
Markov Chains	Functional Information	System Dynamics	Adaptive Networks
Power Law Statistics	Multi-scale Complexity	Dynamical Systems	
Social Network Analysis		Network Models	
		Agent Based Models	
		Multi-Scale Models	

Figure 11 – List of "complexity-appropriate" tools and approaches [70]

The list is divided into four categories. Methods for synthesis are outside the scope of this thesis.[131] Analysis tools serve to extract information from a set of data in order to draw qualitative conclusions. ***Data mining***, in particular, is the umbrella term for techniques that aim to "make sense of large amounts of … data, in some domain" [161]. Of the analysis tools and techniques listed, only nonlinear time series analysis (in a loose sense of the term) will be used in this work. Despite being omitted, Artificial Neural Networks (ANN) present an opportunity for discussing two key issues. First, it is known that an ANN provided with compact inputs can fit any continuous, bounded function up to an arbitrary accuracy [162] [161]. In other words, for most engineering applications, an ANN can map any set of input data to any set of output data regardless of whether or not those two datasets are actually related. Thus, a challenge in emergent behavior identification is to somehow avoid making spurious causal inferences. Second, a special class of ANN called autoencoders are

[131] For a reviews of DSM and other architecture methods such as RAAM, IRMA, and ARCHITECT see Section 1.4 and its references.

routinely used to for ***data compression*** and ***dimensionality reduction*** [163] [164]. In this process, a data set is provided that relates a set of input variables to one or more outputs. If the variables are not independent (i.e. there is some kind of regularity or straightforward dependence between variables), it is possible for the autoencoder to represent the initial data set using a smaller data set constructed with fewer variables (i.e. the space complexity of the data will decrease). Thus, data compression, dimensionality reduction (model space complexity reduction), and changing description length are all analogous operations, which explains why complexity is so prevalent under the Diagnosis category. However, what an autoencoder (or any other method) cannot do is interpret data compression to signify emergence. In fact, such an association ought to be impossible because doing so merely identifies the presence of patterns, rather than the formation of new classes of objects (see CHAPTER 4). Nevertheless, reducing model space complexity forms part of the basis for the arguments in this thesis.

Figure 11 also lists multiple simulation techniques (under Model). Few of these techniques apply to the case studies that will be presented in this thesis. Uncertainty modeling is unnecessary because the simulation will not contain any pseudo-random numbers and this thesis will not study the propagation of uncertainty in simulation outputs. Virtual immersive modeling and feedback control models do not apply to this work, which will passively observe the evolution of a simulation. Functional / Behavioral models can be ruled out for reasons discussed in Sections 1.5-1.7. Game Theory [165] studies phenomena in terms of goals and the way entities achieve those goals. In general, however, emergence need not be goal oriented. Multi-scale models (e.g. [109] [166]) and Network models [154] cannot be used in this thesis *a priori* without begging the question. Cellular

Automata [167] only apply to grids of regularly spaced objects, which cannot be used for SAR simulation.[132] Dissipative systems [168] tend to refer to systems governed by thermodynamic equations, which will not be of use here. This leaves System Dynamics and Agent Based Modeling as possible candidates. Ventana Systems Inc., a leading SD software development company, defines System Dynamics (SD) as “the study and analysis of dynamic feedback systems using computer simulation” [169]. Sayama defines ***Agent Based Models*** (ABM) as simply “computational simulation models that involve many discrete agents” (here, an agent is equivalent to a system component) [154]. An argument can be made that, with the right coding and selection of time step size, both SD and ABM can simulate many of the same kinds of systems. However, Systems Dynamics tends to emphasize graphical representations of properties[119] and their causal links (the objects are abstracted out [170]), while ABM simulations tend to emphasize graphical representations of the objects themselves (the properties are hidden from view). ABM will be selected because (1) it is typically easier to vary the number of components, (2) it facilitates human verification of spatiotemporal patterns of component behavior and organization.

2.2.2 Emergent Behavior within the Scope of Simulated SoS

The studies cited in CHAPTER 1 typically defined system and SoS from a traditional SE or SoSE standpoint. However, Navy Fleet Synthesis studies are, first and foremost, models and simulations. Thus, the challenges of system definition and system/SoS distinction must now be explained within the context of M&S. M&S introduces five idealizations that generally cannot be provided by real systems. The first

[132] One noteworthy example of emergence was the implementation of a Turing Machine using the Game of Life [299] (see discussion in [295], and the Appendices).

four idealizations are that the components of the system can be (1) simple, (2) indivisible, (3) persistent, and (4) predictable. The fifth idealization (5) is that the environment the components interact with is as well understood as the system components. Most questions of model or simulation validity can be traced back to these five idealizations.[133] These idealizations stem from the fact that, within the context of the simulation itself, the only properties and rules that exist are the ones provided by the simulation's author,[134] and so, the simulation becomes a microcosm wherein every behavior of interest that could possibly exist *can* be observed.[135] Since every component of the system is coded into the simulation by design, it follows that every behavior of every component of the system *can* be attributed to the system (even if that behavior is deemed to be a failure mode, trivial inactivity, etc.). As the remainder of this subsection will show, the first four idealizations are properties of the simulated components only. Since the system does not (necessarily) inherit these five idealizations, the emergent behavior of a simulated SoS[136] possesses the same intrinsic types of uncertainty as the emergent behavior of a real SoS. The sources of the uncertainty are fewer and different than those in nature. The sources of instability are different than those in nature. However, there do exist instabilities (just like physical experiments) and sources of aleatory and epistemic uncertainty (just like physical experiments). Figure 12 shows that as the levels of organization increase, the simulation becomes realistic because real systems lack all idealizations and possess all forms of instability/uncertainty. This argument is important both to the ontology presented in this thesis, as well as to the utility of numerical simulations for studying emergent behavior. The goal of this section is to

[133] The other instability/complexity related issues were discussed at the beginning of this section.
[134] As well as those of the physical machine implementing the simulation.
[135] Of course, exceedingly rare behaviors may never be observed. Nevertheless, they can be.
[136] Made up of systems, as per the model-based definition.

explain why the method developed in this thesis can be extended to real SoS despite the constraints imposed by Assumptions 1 and 2.

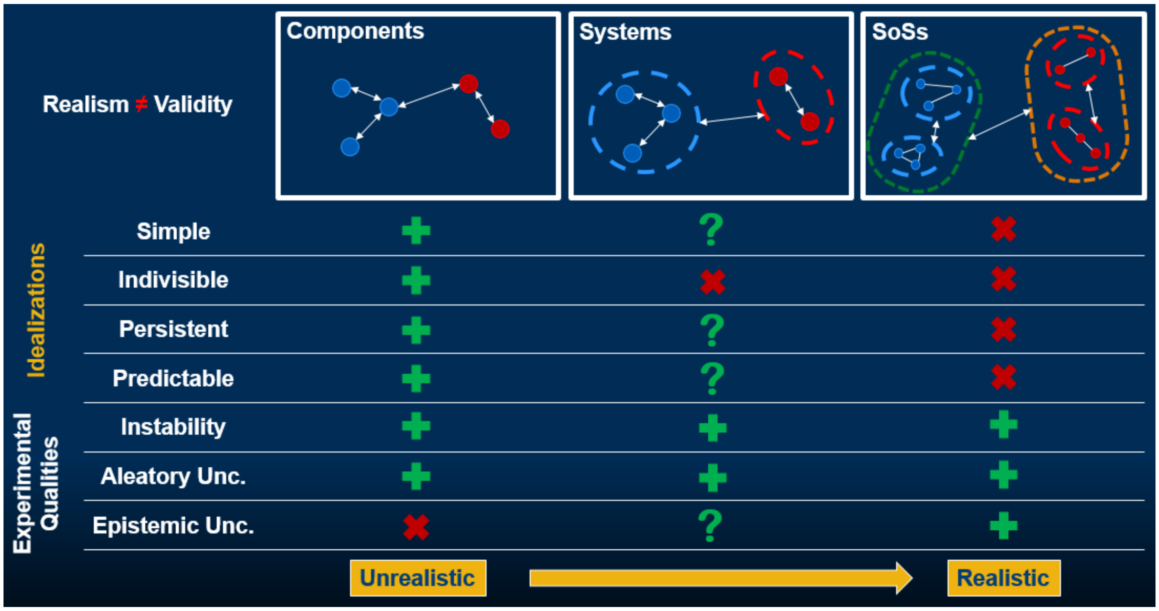


Figure 12 – Progression of realism: As interactions (white arrows) lead to self-organized systems and SoS, the systems either (+) possess, (?) may possess, or (×) do not possess idealizations and forms of uncertainty.

Most authors associate the term simplicity[137] with systems. The qualities attributed to simple systems include:[138] (1) every individual effect has a unique cause; weak causes have small effects; predictability (see [59] and its references), (2) linearity; predictability;[139] decomposable properties; clear cause-effect relationships (see [171]), (3) straightforward physical and functional decompositions containing few elements [81], and finally, (4) a small number of components with negligible interactions; easily isolated from its observational process and environment [80]. These definitions contributed to the

[137] Keep in mind McEver's statement (cited in Section 1.5) that the opposite of complex is not simple, it is decomposable.

[138] These are enumerated by source in order to present disagreements as well as nuanced variations on a theme. Repetition is intentional.

[139] The paper reads "behavior can be predicated," [173] which appears to be a typographic error.

definition of Simple System in Balestrini-Robinson's Complexity Matrix [81], which is a straightforward and useful classification scheme for real-world systems. In the context of this thesis, however, the definition of complex behavior and emergent behavior already account for these characteristics. Thus, simple must take on a different meaning that facilitates the discussion of emergent behavior in a simulation. In this thesis, ***simple*** is taken to mean that all the *component's* properties are known in advance and are mathematically well defined. No new properties appear, and no properties disappear, during the simulation. The system can only inherit the property of simplicity if it never exhibits emergent behavior. ***Indivisible*** simply means that the component in the simulation cannot be structurally decomposed into smaller parts. ***Persistent*** means that the component is present throughout the simulation and cannot be deleted or destroyed. ***Predictable*** means that the causes of component behaviors are readily traced to their effects (at the lowest level of the simulation, this is trivial).

Recall that ***aleatory uncertainty*** is the inescapable uncertainty caused by random processes and can be "reduced to a stationary random distribution" [172]. Numerical simulations require that numerical values be approximated up to single or double precision. Two obvious sources of aleatory uncertainty in simulations are the round-off error of irrational numbers, and the truncation of certain rational numbers. The interested reader is referred to [173] and Chapter 4 of [159] for more information. These sources of error are intrinsic to simulations, just as randomness is intrinsic to quantum mechanics, and thus serve as a genuine form of aleatory uncertainty in simulations. ***Epistemic uncertainty*** can take on several nuanced meanings, but here is simply an error in the model due to simplifying assumptions used in forming the mathematical equation, or the omission of important terms from the mathematical equation due to modeling difficulties [174] [172]. Systems Engineer Matthew Squair, writing about uncertainty in the context of decision making and risk management, argues that, "Complexity in and of itself is not a direct cause of accidents but what complexity does do is breed epistemic and ontological uncertainty" [174]. Within the context of simulations, there is no epistemic uncertainty at the level of the components in the simulation (due to Assumption 1 and the above simplifications). However, components will self-organize into systems, and systems will self-organize into SoS. Since the SoS no longer possesses the idealizations available at the lowest level of the simulation, any attempt to model the SoS using a set of mathematical equations will face all the usual challenges one faces in reality. For example, if the SoS exhibits emergent behavior, is it not predictable, and a model that does not account for that behavior is over-simplified, which is a source of epistemic uncertainty.[140] If the SoS is divisible, which it is by definition, then the model must account for every possible configuration of parts or it will fail to accurately predict behaviors. This source of epistemic uncertainty is particularly bad for fleets of ships, and modular ships. If the SoS contains complex systems, it is not simple, which is another source of epistemic uncertainty on par with unpredictability. If the SoS is not persistent, then there are time intervals during the simulation where the SoS model is completely invalid, which is another source of epistemic uncertainty. In her PhD thesis, Dr. Diana Talley wrote extensively on the quantification of uncertainty in SoS models. One of the gaps addressed in her thesis is that "Existing SoS Design Methods are incapable of modeling all of the different types of relevant uncertainty" [175]. It is

[140] Squair [176] and Beven [174] leave room for debate as to whether this falls under the category of ontological uncertainty. The scope of this thesis does not require the distinction to be clarified, and can proceed without loss of generality. Those that associate "surprise" with emergence will find an opportunity for research here.

important to note that the uncertainty she addressed is caused, in part, by the fact that models of real SoS *must over-simplify* their components (e.g. a model of a county-wide transportation system cannot possibly account for the alertness of every driver, the wear-and-tear on every car, the precise location of debris on the expressway, etc.). Therefore, there is a point where SoS models cannot accurately capture and decouple the effects of aleatory uncertainties and epistemic uncertainties. In an idealized, bottom-up simulation, such as those in this thesis, they *are* decoupled at the lowest component level by design. Only aleatory exists at the lowest level, while epistemic uncertainty grows with scale. This thesis does not argue that the results of a simulation will be identical to reality, because that is impossible. It does argue, however, that once enough levels of self-organization appear in a simulation, there exist sources for the same kinds of uncertainty one faces in reality, and the emergent behavior observed in the simulation is of the same quality as that observed in reality. Thus, the methods used for studying emergence in a simulation can be extended to physical experiments.

> Simulated SoS are not idealized objects. Their simulated behaviors contain sources of instability, aleatory uncertainty, and epistemic uncertainty. Therefore, a method developed for simulations of SoS emergent behaviors can be extended to empirically observed emergent behavior.[141]

Readers interested in making the simulated behaviors closely mimic those observed empirically (including emergence, by extension) are referred to Patrick Meyer's forthcoming PhD thesis [176]. Readers interested in methods for reducing uncertainty in

[141] An aphorism on why this is important: if simulated emergence is not bona fide emergence, then discrepancies between simulations and reality are not errors or simplifications, they are fantasies.

SoS design are referred to Dr. Talley's thesis, the work by Beven and Squair, and the following optimization studies [177] [178] [179].

The fifth idealization, that the environment is as well understood as the components, is the most restrictive by virtue of the sheer number of objects and phenomena it neglects (including behaviors of the environment, sources of instability, and sources of aleatory uncertainty). Real SoS behavior is often difficult to predict due to unforeseen influences from the environment [68] [103] [69] (also known as context dependence [59] [180]). One important consequence of an idealized environment within a simulation is that the number of levels of object-environment interaction is limited by the properties assigned to the simulated environment (perhaps inestimably). For example, if the environment in the simulation is empty Euclidean space, then there exists only distances along some axis, time, and the objects that exist within that space and time. Every object-environment interaction must be measured in terms of distance and time. The reader is referred to the Appendix for additional discussion of the topics presented in this section. Specific examinations of context-dependence have been scoped out of this thesis.

2.2.3 Response to a Relevant Polemic

Professor of Epidemiology Joshua Epstein argues that the whole-parts dichotomy favors reductionism. "Typical of classical emergentism would be the claim: No description of the individual bee can ever explain the emergent phenomenon of the hive. How would one know that? Is this a falsifiable empirical claim, or something that seems true because of a lax definition of terms? Perhaps the latter" [181]. He then proceeds to argue that the characterization of an individual bee is incomplete without also including all of the bee's interaction rules, “It makes little sense to speak of Joshua Epstein devoid of all relationships with family, friends, colleagues… My ‘rules of social interaction’ are, in part, what make me me… When (as a designer of agent objects) you get these rules right… you get the hive too” [181]. Nevertheless, in saying “[when] you get these rules right,” Epstein seems to actually support “classical emergentism”: the interaction rule is the real object of interest, not the bees. Interactions require at least two parties. To properly define an interaction rule, one must fully define both parties involved. The interaction between a bee and a wasp is only one vanishingly small part of what makes a bee a bee (to say nothing of wasps), and the rule itself belongs equally to the bee and the wasp. One could certainly write a computer program where a virtual wasp attacks “anything” but that behavior would not resemble reality. The wasp must target a bee, and the bee must evade or counterattack the wasp, and the manner in which these behaviors unfold is specific to that pair. They are not merely attacking “something.” The wasp acts in accordance with the properties of the bee, and vice versa, and those coupled decisions dictate the interaction rules. Furthermore, if it is important to avoid vague definitions of bees, it is equally important to avoid vague definitions of hives. Wasps also build hives, human beekeepers build hives, and Epstein’s virtual creatures also build virtual hives. As stated in Section 1.4, the mapping between physical and functional decompositions is often not one-to-one. In order to safely neglect the fact that bees are not the only hive-builders and that hives come in many forms and can be built of many different materials, one must either be so specific about which bee(s) and which hive that the model no longer generalizes beyond that one case, or one must be so abstract that the model never fully or accurately characterizes any hive, anywhere.[142]

From a modeling standpoint, such arguments against emergence rely on up to four implicit assumptions: (1) all simulated bees are identical, so that knowing the property of one bee means knowing the properties of all bees, (2) bees have a small number of properties with which to interact, (3) only a small subset of properties are relevant to hive-building, and/or (4) possessing the properties relevant to hive-building implies that the organism possessing them will build hives. The first assumption only matches well with reality when the objects are incredibly simple (perhaps quarks). For engineering applications, the assumption is false since every physical individual is unique, and those distinctions bring important nuance to each interaction (i.e. the "rules" for each interaction are slightly different).[143] Furthermore, just as hive-building is not unique to bees, many other interactions are not unique to bees (e.g. mating, pollination, self-destructive combat, etc.). Therefore, any classification of any object based on a subset of its physical and functional decomposition is inherently vague. The second assumption is only made by scientists and engineers for the purposes of simplifying problems to make them mathematically tractable, and in doing so, neglect the holism-reductionism debate altogether. In reality, once a model moves beyond quarks or neutrons, that assumption rapidly becomes invalid. Similarly, the third assumption is only approximately valid in an abstract sense. The thing one obtains by relating some properties to some behavior is an

[142] Yet another example: the physical forces that cause spiral galaxies to exhibit the golden ratio are not the biological interactions that cause nautilus shells or ferns to exhibit the golden ratio. One cannot use a model of gravitation to generate the nautilus shell. One cannot use the model of a bee to produce a wasp hive.
[143] Engineers typically handle this with part tolerances, design margins, and other risk mitigation techniques.

approximate model of a mathematical representation of the behavior, not a physical instance of the behavior. Many things conform to that model (as stated earlier) but a model is a model, not a bee or a hive. Therefore, the modeler must be willing to tolerate a certain amount of error in the simulation. Enumerating the rules of interaction for a real system places an insurmountable burden on the modeler. For example, interaction rules of a bee vary with its age. A larva does not interact like an adult, and there are multiple stages of maturation to consider. Developmental issues are also important, as seen in human populations, because they can significantly impact health, which affects interaction rules.[144] Well-written interaction rules also require that all parties involved be sufficiently well defined. Aside from the fact that each real bee-bee interaction is unique, there are also interactions with different types of bees (e.g. competitive worker-worker-queen mating rituals, cooperative worker-worker building all of which must occur for real hives to exist), or the interaction with a different species such as predatory wasps, or humans. Then there are interactions with the rest of the environment: solar radiation, dust, viruses, bacteria, toxins, the forest, the weather, etc. If those entities are brought into the simulation, one must characterize their rules as well, and so on, until it becomes necessary to model the entire physical universe in order to faithfully model any single real-world hive (see Section 3.2). Finally, the fourth assumption is clearly an oversimplification. Just because a person can build a beehive does not mean that said person will ever build a beehive in their lifetime (i.e. sensitivity to initial and boundary conditions / the environment).

[144] For example, the pregnant women suffering from famine in the Netherlands during World War II bore children that experienced higher rates of obesity, diabetes, and schizophrenia (clear impacts on the interaction rules of their children) [180] [181].

2.3 Defining Property, Behavior, and Interaction

Since SoS often involve human or autonomous decision-makers, some researchers draw a distinction between a behavior and the quantifiable properties assigned to a system component. Especially with regards to human decisions, studies often focus on whether or not the rules used in the simulation accurately reproduce a particular behavior or decision-making process. These concerns are eliminated by Assumption 1 and the fact that machine learning techniques will not be used in the ABM presented in this thesis. Therefore, 'the decision to behave in manner *x*' cannot be considered a behavior (as strategies are in [182]), because it is dictated by rules assigned to the component at the creation of the simulation.

In the context of this thesis, a ***property***[119] is simply any model variable that represents a quantity used to uniquely characterize a specific component, system, or SoS. At times, different objects may possess properties equal in value, such as two bricks having the same mass. This thesis will focus on real-valued properties. A ***behavior*** is simply the time-rate-of-change of a property.[145] A ***relative property*** or ***relative behavior*** is simply the property or behavior of an object X taken with respect to another object Y. For example, if p_x and p_y are scalar properties of X and Y respectively, then the property of X relative to Y, $p_{x|y}$, is given by the equation $p_{x|y} = p_x - p_y$. Other mathematical conventions (such as dot products for the component of one vector relative to another, or the equation for conditional probability) are also permissible for relative properties or behaviors. The subscript for relative property will remain consistent regardless of the operation used to calculate that relative property.

An ***interaction*** is the change of one object's properties or behaviors due to the property or behavior of another object (see similar definition in [139]). Some authors favor

[145] Generalizations to gradients in space or other coordinate systems are outside the scope of this thesis.

implementing conditions that suggest interaction (such as proximity) rather than directly accounting for interactions because interactions may be impossible to discern if too many occur in rapid succession [183] and may be used here when appropriate. A ***direct interaction*** is defined as the change in the property of one object exclusively as a function of the properties/behaviors of another object. If systems X and Y have properties p_x and p_y respectively, and t represents time, then ${}^{\Delta p_x}/_{\Delta t} = f(p_y)$[146] is an interaction (the property of Y is changing the property of X), and ${}^{\mathrm{d}p_x}/_{dt} = f(p_y)$ is also an interaction. An ***indirect interaction*** refers to a three-object cause-effect chain where one object interacts with an intermediate object, and the intermediate object interacts with a third object (see Kim's discussion of ant pheromone trails [59]).

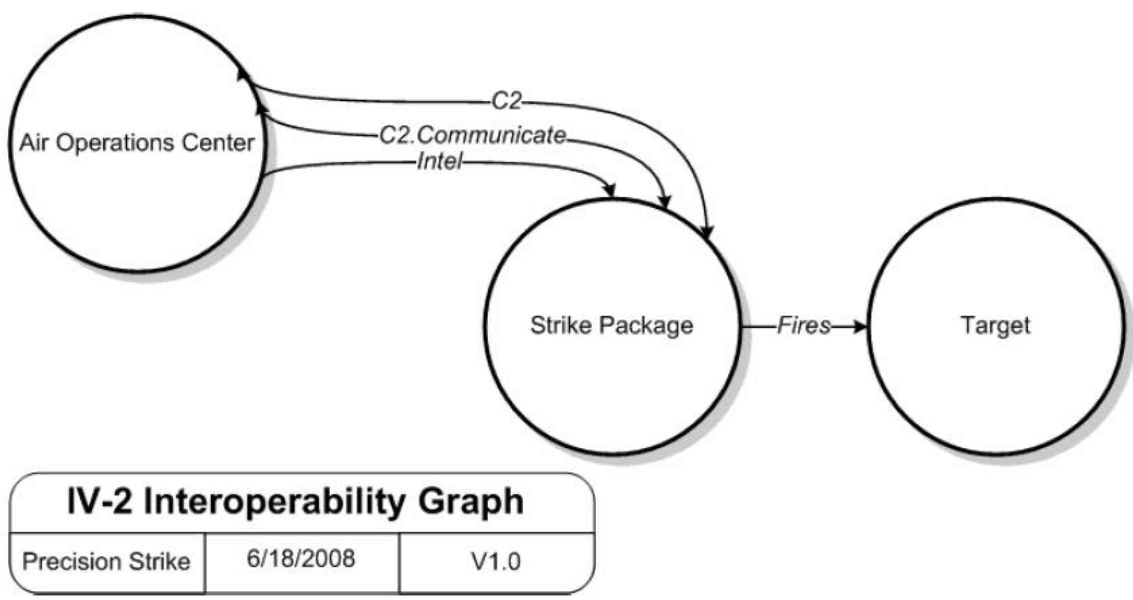


Figure 13 –Directed multi-edge graph of precision-strike behavior: Ford's proposed Interoperability View [26]

These definitions of interaction are compatible with the subcategories of interoperability defined by Ford [26] (see footnotes 42, and 89). With respect to Ford's

[146] The symbol Δ is taken to mean the change in a property, as per mathematical convention.

taxonomy, the direct interaction defined above is an example of confrontational interoperability (the effect of one object's properties or behaviors is imposed on the other), and would be represented using a directed graph as in Figure 13.[147,148] In Ford's graph, for example, the strike package is directly interacting with the target. Modeling collaborative interoperability, or confrontational interoperability in the absence of a clear advantage (i.e. the target returns fire), would require a set of direct interaction equations that would couple the changes of properties and behaviors of multiple objects over time. Finally, note how Ford's graph refers to the set of interactions required for a higher-level capability (Precision Strike). This is an explicit example of the need for a hyper-graph (see Appendix). Furthermore, since the C2 edge is bidirectional, there is, in fact, a cycle in his directed graph, as discussed in Section 1.6 (this cycle suggests an iterative process where the Strike Package and Air Operations Center sequentially send each other messages, which is related to the definition of a feedback loop, but is not quite the same phenomenon).

2.4 A Notional Adversarial Model

Combat simulations are notoriously difficult to validate [153]. Encoding even a fraction of the guidance contained in the better known aerial combat manuals [184] [185] into the decision logic and physics of an ABM would be a PhD thesis unto itself. Nevertheless, at its core, combat is a competition, which is easy to simulate at an abstract level. Recall that the purpose of the CBA is to recommend whether a new physical technology or platform needs to be purchased (means), or a change needs to be made to the military's tactics, strategy, doctrine, etc. (ways). FSS then take these decisions, or different combinations of such decisions over time, and extrapolate those decisions 10-30

[147] Ford's proposed "view" refers to documentation relevant to DoDAF. See [287] for more information.
[148] Note also the feedback loop between Air Operations Center and Strike Package via the two-way C2 interaction. Feedback loops are a recurring theme in emergent behavior literature.

years into the future. A properly coded adversarial mission model is a straightforward mechanism for replicating the ways-versus-means recommendations made by a CBA at a conceptual level, since "winning" the mission can be achieved either through better decision-making behaviors on the part of the simulated combatants, or by improving the material properties of a simulated team. Finally, all combat involves self-organization,[149] which is a crucial component to emergent behavior (see Section 1.7 and 3.2). Therefore, this thesis will implement an adversarial model inspired by WWII-style dogfighting. More details about this ABM are discussed in Section 5.7and CHAPTER 7.

Colonel John R. Boyd was an American fighter pilot that fought in the Korean War and was also an influential military theorist.[150] In addition to authoring a widely-studied manual on aerial tactics [184], he continuously developed over the course of his career a series of briefings titled "A Discourse on Winning and Losing" [186]. This and several other documents, as well as links to YouTube videos of Boyd's lectures have been compiled into one pdf document by Dr. Grant T. Hammond in such a way that make their historical importance is readily observable. Col. Boyd is also well-known for his description of human and organizational decision-making, which he called the Observation, Orientation, Decision, and Action loop (OODA loop, see Figure 14). Consistent with the aforementioned discussion on functional decompositions of emergent behaviors, Boyd's OODA loop is a cyclical, directed, layered hypergraph.[151] "The OODA

[149] Short range combat requires the obvious, short-term positioning and interactions of combatants, while long range combat (such as ICBM attacks/defenses) typically involves the long term positioning of stationary combatants.
[150] This is an understatement. The interested reader is referred to [188] [323] [189] for more information about his extensive career, as well as the list by Franklin Spinney [324].
[151] Boyd's personal collection of papers and books [187] suggest he read a fair amount of literature on complex systems and emergent behavior, including on General System Theory.

Loop was a nonlinear process with constant feedback and feed-forward channels of implicit guidance and control," [186] which Boyd himself described as "an evolving, open-ended, far from equilibrium process of self-organization, emergence, and natural selection" [186]. Like dogfighting, incorporating such a sophisticated decision-making framework into an ABM would be another research effort altogether.

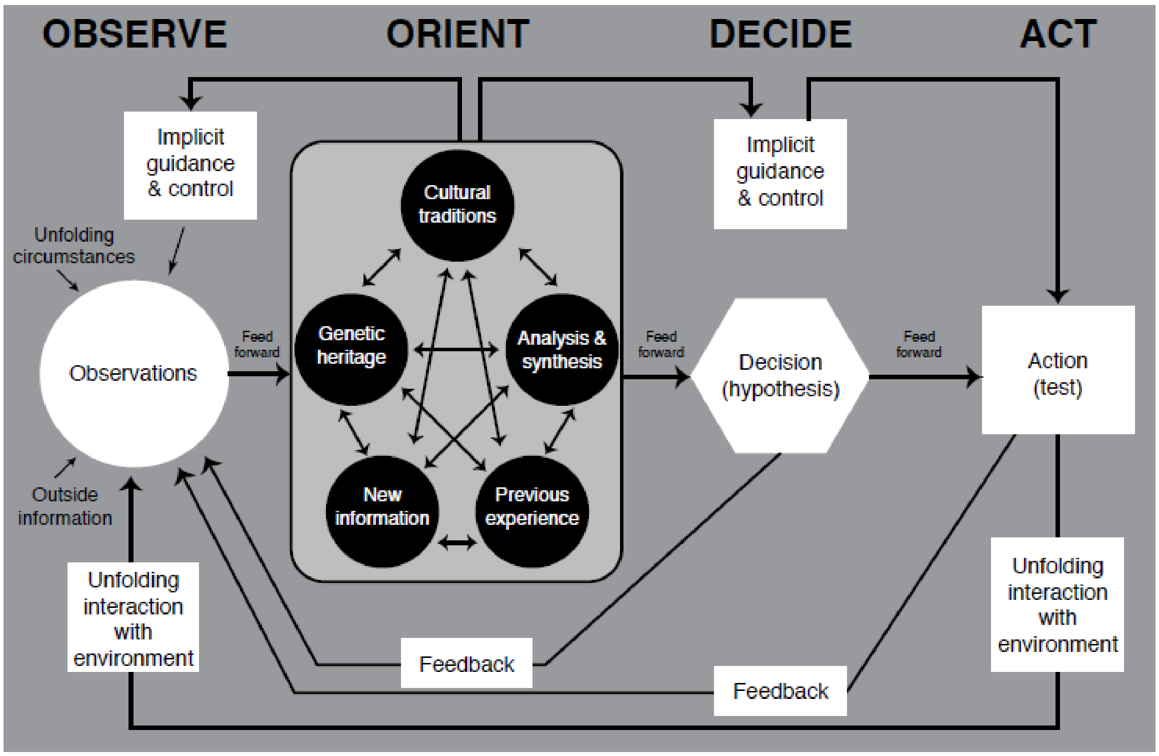


Figure 14 – John Boyd's OODA Loop, reproduced from [187]

This example, as well as the interoperability figure by Ford (Figure 13) show that complex behaviors can be depicted using similar graphs whether they occur in a decision-making context, or a design context (as shown in Figure 8). The topic of graphing complex behaviors will be revisited in Section 5.3. Regarding Boyd's Discourse [186], the discussion of patterns as well as self-organization and emergence is a recurring theme in all literature on complexity. While Boyd was concerned with the application of these ideas to war fighting and policy making, this thesis will examine them in a strictly mathematical sense in CHAPTER 3.

CHAPTER 3. PATTERNS AND EMERGENCE

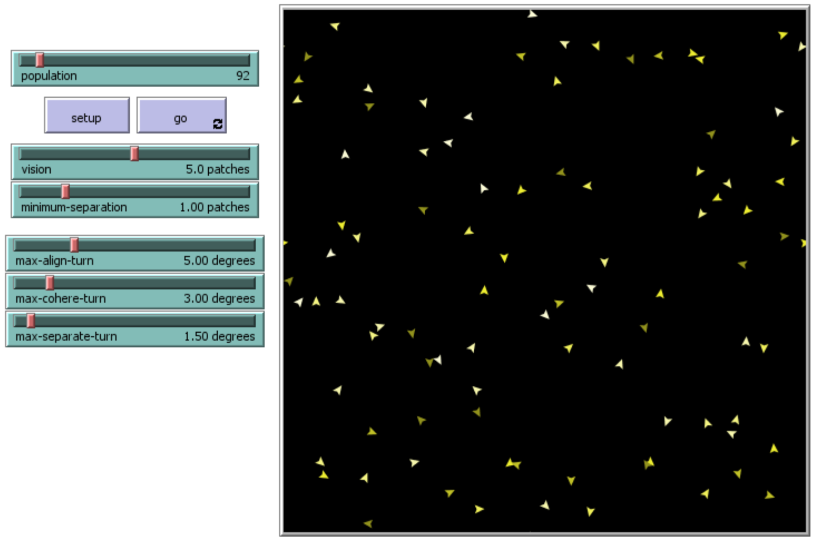
A simulation often cited in the literature on emergence, now commonly referred to as the Boids Model, was first developed in 1986 to show that the flocking behavior of many species of animals could be simulated using an ABM using a small number of rules and very simple components [188]. The NetLogo implementation of that simulation is referred to as the Flocking Model [189]. Under the right settings, and given enough time, the simulated components often coalesce into a group and subsequently proceed to fly in the same direction.[152] That is, the simulation converges to a stable configuration that is typified by components possessing a nearly uniform heading, as well as a relatively persistent arrangement in space. See Figure 15, where (a) is the initial random distribution of components, depicted using small arrowhead icons, and (b) is the distribution of components after 2,898 time steps. It is worth noting that in this particular simulation, over the course of several thousand more time steps, small groups of components would occasionally break off, travel independently, and then merge back with the larger group (the simulation was not perfectly converged/stable).

Although authors discussing this model generally agree that it depicts an emergent behavior (notably, Phelan refers to this emergence as an illusion [170]), authors disagree over which specific transient features to refer to as the emergent behavior.[153] A convenient feature of the Boids Model is that all interactions occur once one component enters another's vision radius.

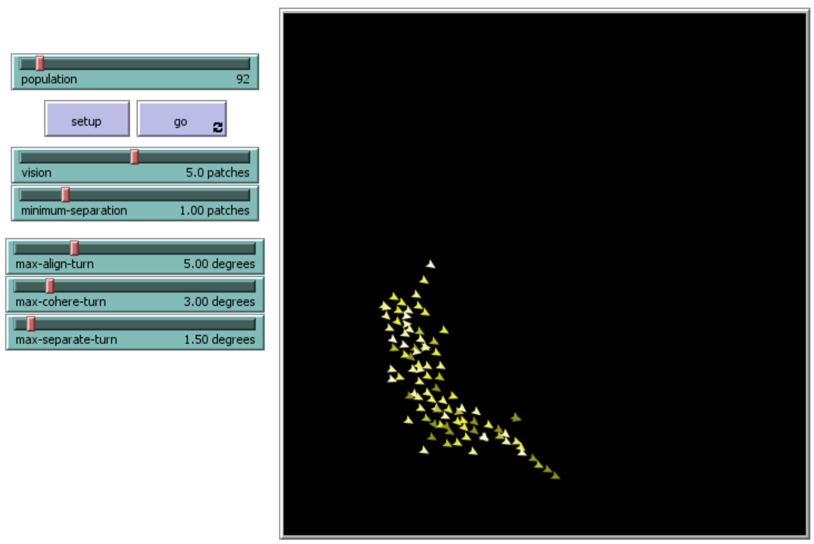
---

[152] Subject to instabilities that cause some components to oscillate with respect to the group's trajectory, or eject from the group.

[153] Table 2 in [97] lists papers that contain examples of emergence. The authors list, but do not cite, a paper by Miner [289] (perhaps they meant to remove it from the paper). Miner's work is in relation to data visualization. Miner does not propose density as a metric of emergence.



(a)



(b)

Figure 15 –NetLogo GUI and flocking simulation (a) before and (b) after ~3,000 iterations

Thus, Moncion, Amar, and Hutzler capitalize on the rule set of the Boids Model to utilize proximity and relative heading as indirect measures of interaction [183], bypassing the need to explicitly relate one or more behaviors to one or more interactions. They only refer to flocking as an emergent behavior, but their narrative suggests that spread/density of the flock could be one property, and number of components could be another property, thereby classifying splitting/merging as a flock behavior. Szabo and Teo [96], Chan [139], and Seth [142] refer solely to flocking as the emergent behavior. Choate refers to swarming itself as an emergent behavior [190]. Swarming can be distinguished from flocking in that obstacles avoided by one component are subsequently avoided by all components (i.e. it places more of an emphasis on collective maneuvering). Phelan also referred to collective obstacle avoidance, as well as "wheeling" [170].[154] In every case, synchronized behavior, patterns of behavior, or sudden disruptions of patterns are the common themes underlying the suggested emergent behaviors. This research will not go beyond the self-organization (Section 3.2) typically associated with the model. The remainder of this chapter will discussed the basis for this data compression and its relationship to emergent behavior.

3.1 Pattern Recognition

Several authors writing on the subject of emergence cited in this thesis make an explicit or implicit reference to patterns as an underlying feature of emergence. For example, the Complexity Primer specifically advises engineers to "Identify and use patterns… Patterns are the primary means of dealing specifically with emergence and side effects," [61]. Minati's research on emergence centers on the notion of a ***collective system*** which is "established by permanently interacting generic agents provided or not provided with cognitive systems,"[155] [191]. Minati then argues that collective systems have what are called spatial collective behaviors (behaviors attributable to the system resulting from component interactions), and inherit properties (called meta-structural properties) from, among other things, any regularities including periodicity, quasi-periodicity, and chaotic regularities.[156] Anderson's discussion of symmetry is, by definition, a discussion of

[154] Wheeling is not described in the text.
[155] Minati's definition of interaction is compatible with the one adopted here. Figure 1 in [290] is a clear and simple depiction of collective systems, and corresponds to the discussion on hierarchy and heterarchy in CHAPTER 1.
[156] This argument will be extended in CHAPTER 4

regularities and patterns [110]. Polack and Stepney associate patterns in time as well as space with emergence [115]. As a final example, although information entropy is not considered in this thesis, Szabo and Teo cite a variety of sources that show information entropy changes have been associated with the onset of pattern formation in systems exhibiting weak emergence [96].

Pattern Recognition is a term broadly used in machine learning [161], but in this thesis, it will take the very specific meaning of identifying periodicity in a data set for two reasons. The first reason is that the mathematics is intuitive, and the tools for this approach are well established and widely available. To identify periodicity simply means that some measurable phenomena (e.g. a property of component X, p_x) can be modeled using a periodic function or an almost periodic function [192]. For example, if p_x is periodic in time, an equation can be written $p_x = f(t)\ s.t. f(t) = f(t + c)$, where *c* is some constant. Since empirical data sets are finite and the goal is effectively to fit a function, *f*, to that data it is generally impossible to analytically prove that *f* is the true function representing the behavior. Just as an ANN can essentially fit any data set, a polynomial or a Fourier series can be used to fit most practical data sets.[157] Nevertheless, a compelling argument can be made that a data set is likely to be periodic if two conditions are met: (1) the data can be fit with a Fourier Series after a certain number of periods, n_r, and the prediction error of that Fourier Series remains bounded beneath a strict threshold as the pattern persists for a number of periods $n > n_r$,[158] (2) the coefficients of that Fourier Series converge to zero at a particular rate [193]. Although, neither of these conditions constitutes a proof, they are based on well-established approaches for which proofs do exist. Unfortunately, the assumptions required for those proofs cannot be taken for granted because mathematically

[157] One can assume data corresponding to smooth and continuous functions without loss of generality.
[158] This would suggest that the pattern converged and the current Fourier series is a good-enough approximation.

“nice” conditions[159] typically only appear in the absolute simplest ABMs. This means that Fourier Series can be used as a good indication of periodicity, but false positives are possible.

The second reason the identification of periodicity is critical in this thesis is that, unlike emergence, there is much less controversy regarding the ontological claim that “patterns exist” (though there is some [194]). Thus, if a persistent pattern of interaction, or a pattern of relative behavior present among a collection of components can be identified and modeled, the only remaining ontological step is to show that the collective is an object unto itself (a collective system, higher-level abstraction, SoS, etc.). Within the context of a simulation, this means that, at a minimum, the object can be meaningfully represented using a set of properties and behaviors. It follows from the in 2.2.2 argument that, within the context of a simulation, only components possessing five idealizations are guaranteed to exist (axiomatically). If a second simulation can be written such that a system is coded as the fundamental component, then it becomes possible in principle to verify the behaviors exhibited in that simulation against the predictions of the original simulation (which is valid by Assumption 1), given appropriate initial and boundary conditions. However, even under ideal conditions, it may not be possible in practice to perfectly verify the second simulation. Work on emergent behavior property identification by Samaey, Holvoet, and De Wolf suggests that as there is a prohibitive diminishing return relationship between the accuracy of low-level initial conditions reverse-engineered from high-level properties and the amount of data required to improve accuracy of those initial conditions [195]. This suggests that, for all practical purposes, higher-level simulations cannot be built from lower-level patterns without some loss of information.

[159] Such as the ability to gather a large enough set of data points to rule out nonsensical overfitting.

One useful feature of patterns and pattern recognition is it can be used for dimensionality reduction and data compression [161]. If two objects are engaged in a pattern of relative behavior, for example given by $p_x - p_y = f(t)$ then rather than simulate both objects using their individual behavior rules in, for example, an ABM, it becomes possible to simply calculate the properties of the second object using its relationship to the first instead. In this sense, the dimensionality has been reduced and the data compressed. Since the property attributed to the simulated object is also a property of the model it was derived from, one can say that the model was compressed. However, since their interactions are what generated the pattern in the first place, deleting one entity would disrupt that pattern. Therefore, it seems reasonable instead to treat the two objects in this example as a collective, and then to select properties for the collective object from which to back-calculate the properties of the individuals as suggested earlier. For example, the Moon orbiting the Earth as it circles the Sun could be replaced with the simulation of a single, rigid, rotating rod whose length is the distance between the center of the Earth and the center of the Moon, that orbits around the Sun in such that the positions of the Earth and Moon can be recreated from the centroid of the rod and its length. The action of gravity between the Earth and the Moon is replaced by a simplified representation due to the stability of the Moon's orbit (the pattern). However, it was also suggested earlier that reverse-engineering the information for the low-level entities can require so many additional terms and/or equations as to make the effort counter-productive.[160]

If the pattern, *f(t)*, is not perfect because the low-level pattern has not fully stabilized, or is subject to perturbations, ε, such that $p_x - p_y = f(t) + \varepsilon(t)$, then the continued use of *f(t)* becomes a simplification and information about ε is lost (i.e. the model relying on *f(t)* is inconsistent with the original ABM). Therefore, the programmer can either make *f(t)* more complicated to account for perturbations, or accept the inconsistencies/information-

[160] This speaks to the incompressibility of the simulation (see Section 3.2).

loss generated by the simplified simulation. Since most of the phenomena presented in this thesis will never reach perfect equilibrium, the data compression will typically result in ***lossy compression*** (as opposed to lossless data compression [161]). Although perfect patterns can lead to lossless compression, in principle, the other nonlinear behaviors commonly associated with emergent behavior (chaos, bifurcations, fractals, limit cycles, etc.) inevitably result in information loss under even the slightest simplification. In these cases, the only hope the programmer has of generating a useful higher-level model depends on whether the chaos, for example, is bounded. If it is, then the error can be bounded, and estimates of how that error propagates through the system can be taken.

3.2 Self-Organization

Self-organization is defined as "the transition of a system into an organized form in the absence of external or centralized control" [196]. Self-organization is essentially the phenomenon of patterns appearing spontaneously within the context of unstable nonlinear models [197].[161] Several authors consider self-organization to be a precursor to emergence [198], or even equivalent to it [199] [200]. Although markedly different definitions and associations also exist (see references in [200], as well as [201] [202]), they are beyond the scope of this work. Numerous examples of self-organization appear in nature [200] [203]. Flocking is one such example.

As discussed in Section 2.2, nonlinear models can produce phenomena such as chaos and bifurcations, or converge to stable solutions as in the case of self-organization. Despite being deterministic, these phenomena are impossible to predict without running a simulation of the model starting from a particular initial condition (each initial condition

[161] Halley and Winkler make a distinction between the emergent properties of equilibrium systems and nonequilibrium systems. Any structure not assembled by some intelligence must have self-organized, whether that system is currently in equilibrium or not. If it is not in equilibrium the structure is said to be ***metastable***. This thesis will not make a distinction between the properties of equilibrium systems and nonequilibrium systems.

can produce wildly different results). Therefore, nonlinearity can introduce an objective form of unpredictability that, unlike subjective forms (Section 1.7), can be used as a necessary condition for emergence [117]. Huneman describes this objective unpredictability of nonlinear models as a form of information incompressibility. He counters Epstein's reductionism on the grounds that since bee interactions must be fully simulated in order to observe the hive, the simulation is incompressible. Huneman argues that it is a stronger argument to associate emergence with processes (i.e. how events unfold) rather than arrangements of objects and the whole-parts dichotomy where properties are ascribed to the whole and not reducible to their parts. Huneman refers to the former as computational emergence, and to the latter as combinatorial emergence. He considers arguments for combinatorial emergence to be weak. However, the two notions are not mutually exclusive. A group of objects can arrange themselves into a self-organized pattern via an incompressible process (i.e. nonlinear dynamics), and upon doing so, exhibit behaviors that the individual objects cannot exhibit in isolation. Huneman argues that compressibility should not be considered a basis for emergence, writing, "according to the computational view it is never the pattern as itself that is emergent" [117]. He also argues that "emergence is a feature of the whole agent-based simulation process" again due to the process by which it occurred [117]. Huneman then adds, "weak emergence defined as inaccessibility except by simulation is thereby not something trivial" [117]. Although Huneman keeps the two concepts apart, this thesis will show that both computational emergence and combinatorial emergence work hand-in-hand.

Since the simulation of the nonlinear model is incompressible up to the point where periodic behavior begins, the onset of self-organization marks a change in the underlying information content of the simulation, and, therefore, should be taken to indicate that *something* has occurred. The question is whether that something is merely pattern formation, or something more (this is the subject of CHAPTER 4). Ryan, in a paper unrelated to Huneman, writes “formal systems, including mathematical models and computer simulations, are incapable of reproducing naissance emergence. This does not mean that once naissance emergence has occurred that we cannot alter our models to [account for it]. It just means that we cannot do it *a priori*, because we require empirical access to select between the possible properties of completely new configurations… [It] is an ontological concept … [and] cannot be epistemic” [111].[162] For now, briefly consider the limitation of Huneman’s emphasis on incompressibility: if the simulation must be run to obtain the result (and it must), then what information does one have left with which to determine that a particular behavior is emergent? Or even that an emergent behavior exists? It would appear that this information would have to be added to the simulation data. Introducing new information to a problem requires additional justification, which is undesirable. Recall, however, that once self-organization appears the system is configured in a stable pattern. The properties of the components engaged in that pattern can now be described using a compact set of equations that are simpler than the original nonlinear set. Since this new set requires fewer variables, the system of equations has been compressed in a sense. If the data can be compressed, then there is a deficit in the information contained in the model relative to the initial setup (loosely speaking). It may be that the new emergent properties of the whole can come into existence because this pattern-induced compression has made information available for novel use.[163] In fact, Ryan argues, “emergent properties must be the result of spatially or temporally extended structures… By structure, we mean there is a pattern that relates the components, which implies redundancy, and therefore the description of the components is compressible.” [111] What we have not yet done is determine what properties or behaviors the collective object has. In order to avoid confusion going forward, Huneman’s incompressibility will be referred to as objective

[162] Ryan defines naissance emergence as “the source of novelty.” The interested reader is referred to Ryan’s paper for more information. The distinction can be neglected here.
[163] At least one other thesis has associated data compression with novelty [331], but apparently in a very different context. A study adapting the Normalized Compression Distance to self-organization is warranted.

unpredictability, and is included in the list of necessary conditions for emergence. We now see that self-organization can come about due to objectively unpredictable processes (the simulation must be executed in order to observe the outcome), but the consequence of self-organization is the compression of information content, and this deficit in information content enables the rise of emergent properties. Clearly then, the type of pattern determines the quantity of data compressed, and the quantity of data compressed could be used as the basis for counting the number of new emergent properties possessed by the self-organized group. This would provide a clear link between computational emergence and combinatorial emergence, thereby resolving the tension in Huneman's paper.

El-Hani and Pihlström refer to the discussion in the previous paragraph as the problem of "[getting] something from nothing" [204]. Their article turns from that problem to the literature that addresses the problem using downward causation. Rather than rely on downward causation, this thesis solves the something-from-nothing problem directly. Between any two models at any two levels of abstraction, information is both lost and compressed. That deficit of information creates the "something" from which new higher level properties can be created. Both models can be simultaneously valid, and perhaps even invertible. It is much more common to find that one model is the limiting case of another as with Kinetic Theory and the Navier-Stokes Equations [205]. Therefore, with respect to downward causation, this thesis simply takes the position that some models are easier to write from the bottom up, and sometimes they are easier to write from the top down (some authors go so far as to say that many cause-effect relations are invertible [198]). The hypotheses in this thesis will not attempt to measure the information lost. An exact "conservation of information" -style equation is left as future work.

Similar to Huneman, theoretical physicist Dr. Eric Bonabeau also drew a distinction between self-organization and "functional" definitions of emergence [200]. Importantly, Bonabeau points out that self-organization alone does not provide a characterization of new functions/behaviors exhibited by the system (hereafter this will simply be referred to as ***functional emergence***). One of the more useful definitions of functional emergence that Bonabeau cites is given by Steels, "[it is a function achieved] indirectly by the interaction of more primitive components among themselves and with the world." Without going into unnecessary details on other variants of the term emergence, [164] Bonabeau writes "the main difficulty [is] to make the link between structure and function"

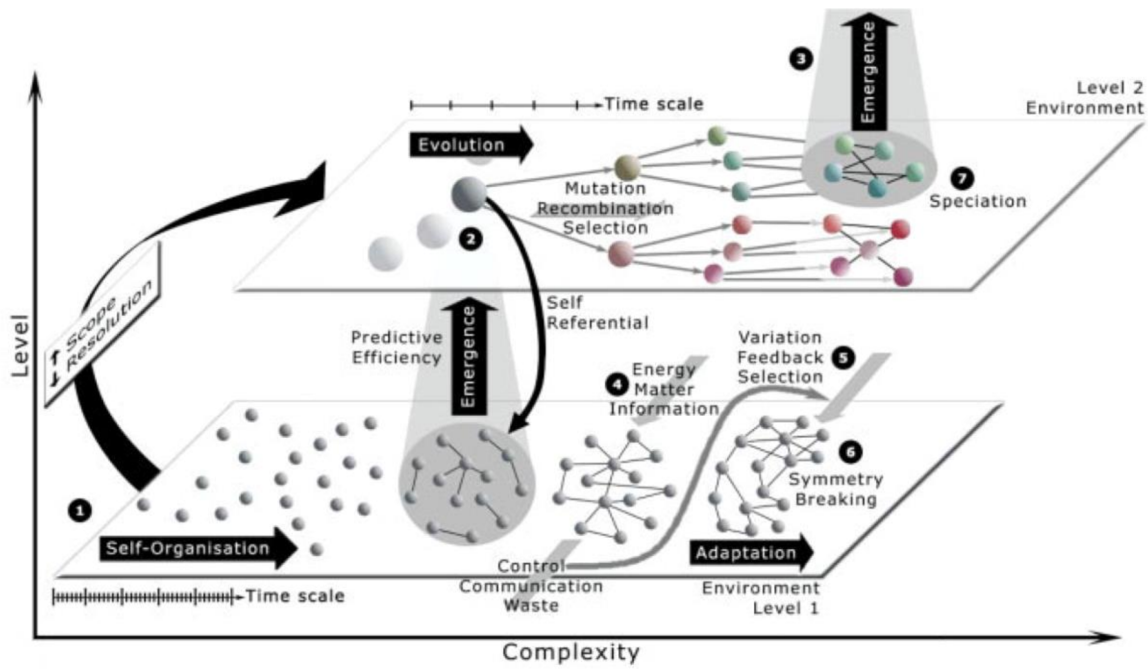


Figure 16 – Graphical depiction of self-organization and emergence in [206]

The notions of weak emergence and functional emergence are the two main concepts to be used in this thesis: a group of self-organized parts will exhibit properties and behaviors attributable to the whole (weak emergence), and although those behaviors can be attributed directly to the whole, they are enacted indirectly by the parts (functional emergence). This

[164] Bonabeau's paper lists 13 examples of various emergent phenomena, and names at least 10 variants of the term emergence including micro-, macro-, nomic, physical, social, biological, psychological, computational, thermodynamic, and semantic. Of those here listed, semantic is the most relevant, but only in that "semantic emergence corresponds to functional emergence" because "in order to model [it], it is necessary to add new observables." [199] That is, functional emergence requires the introduction of new variables to the system of equations.

relationship is depicted in Figure 16, which shows how objects (indicated by spheres) self-organize (lines connecting spheres) into collectives and interact. These collective objects (spheres at a higher level of abstraction) can then self-organize and interact. Each stage is an example of emergent behavior.

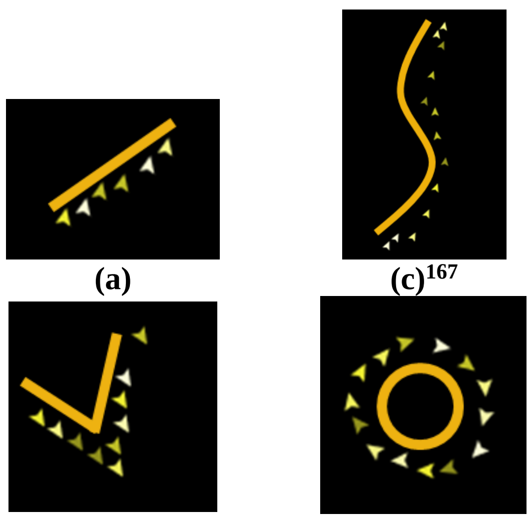
Wright et al. proposes measuring self-organization as a loss in the degrees of freedom of a system in the form of an entropy calculation [207], as do Licata and Minati (although they associate it more explicitly with emergence) [208]. As before, since the measures presented do not enable bridging the gap between form and function (self-organization and subsequent emergent behaviors), they fall outside the scope of this thesis. Prokopenko et al. go much further in providing definitions for self-organization, complexity, and emergence with the aim of defining them such that they are readily distinguished in order to facilitate discussions between researchers in difference disciplines (biology, engineering, etc.) [206]. There are many parallels between their work and this thesis.[165] This thesis largely agrees with their argument that, from an engineering standpoint, the distinction between an emergent behavior and the self-organization preceding it is a matter of computation. As in other research, the authors again rely on information-theoretic measurements, but only as "place-holders" [206] for the mathematical calculations that are clearly needed in any emergent behavior identification method. They admit that "whether this view and these [information theoretic] tools can be successfully applied to [Complex Systems Science] is far from obvious," [206]. Up to this point, this document has dismissed all measurements of emergence, complexity, and now self-organization on the grounds that

[165] There are also crucial differences in definitions, most of which are already covered at least indirectly in Sections 1.6-1.7. Some of those differences are pragmatic (e.g. based on the target audience) and do not warrant discussion. For the purposes of this section, it suffices to say that they also distinguished between self-organization (roughly "pattern emergence") and a form of functional emergence ("intrinsic emergence," after Crutchfield, whose work is also cited in this thesis, but appearing in a different publication [201]).

they fail to bridge the gap between form and function. A detailed explanation of why such methods fail is presented in Section 3.4.

3.3 A Canonical Example of Self-Organization

NetLogo contains a built-in simulation of flocking behavior whose rule set is slightly different from the original Boids model. This simulation, called the Flocking Vee Formations model [209], creates flocks of different shapes using rules that enable the components to accelerate in response to neighbor proximity and, provided it is not too close or too far, implements the rule that the component will adopt the speed and heading of its nearest neighbor (within a restricted cone of vision). Of the patterns visible in this simulation, the four most common are: (a) the Line, (b) the Vee, (c) the Wave, (d) and the Circle[166] (see Figure 17).



(a) **(c)[167]** **(b)** **(d)**

Figure 17 – NetLogo Flock Patterns

[166] This is probably the wheeling behavior Phelan referred to [172].
[167] Better examples of a near-perfect sine wave have been observed, but never with the record button on.

The circle is a persistent configuration and is the second-most sensitive to perturbation. The wave is an unstable, transient configuration (most perturbations destroy the shape). These flocks will be considered a system, and the self-organized patterns of component arrangement will act as sources of information for flock property and behavior definition (e.g. a line has a length, a circle has a perimeter and radius). Flocking, itself, will be treated as a synonym for self-organization. There is room for debate regarding whether anything besides a 2-boid line should be considered a SoS rather than a system (all other shapes are clearly made of 2-boid lines). In general, the number of birds is, first and foremost, a quantitative change in system-level properties, not a qualitative change. In this sense, a n-bird line can be treated as a system without imposing additional assumptions. A Vee, on the other hand, can be modeled as two lines that share a lead bird, which is qualitatively different from a single line, and introduces an additional symmetry to the shape (reflection about the axis formed by the heading of the lead bird, depending on the number of birds). A Circle is a leaderless flock, and has yet another symmetry in its shape (rotational) and requires a minimum number of birds in order to form. In both of those cases, the number of birds takes on additional qualitative significance and present an opportunity for further study.

This is very different from authors that call flocking (or similar coordinated behavior) an emergent property or behavior of the birds. By analogy, the Thach Weave is an example of self-organization, not an emergent behavior of a pilot. The self-organized object is the leader-attacker pair, which the wingman is trained to intercept and destroy via the Thach weave. In this case, as long as the self-organized pattern holds (the attacker pursues the leader), then the wingman can predict the heading, speed, and separation of the pair (therein lies the emergence), and destroy the pair by firing at the attacker, which will presumably disengage, or be destroyed.[168] If the attacker is not destroyed, or does not disengage, the pair has not been destroyed and the maneuver failed. In addition to the discussion in Section 3.2, detailed reasoning is provided in Section 4.1 to justify why flocking is not an emergent property/behavior of birds according to the terminology in this thesis. The properties of flocks (center of gravity, etc.) may be emergent properties of the collection of birds (subject to the conditions in CHAPTER 4). A bird cannot have an emergent behavior (at the bird level of abstraction), although many of the properties of birds are emergent properties with respect to its organs (e.g. consciousness). The same idea will be extended to pilots in combat. Pilots engaged in one-on-one dogfighting would not perform the weave maneuver, since there is no pair to destroy (they would use some other tactic to anticipate/manipulate enemy behavior).

Referring back to the flock examples, the geometric patterns exhibit easily identifiable patterns in component relative heading, relative acceleration, and relative distance, which facilitates verifying that a Fourier Series fitted to the data corresponds to an actual pattern.[169] Consider the case of a line of six boids, depicted in Figure 18.

[168] Aside: this is where training artificial intelligence easily goes wrong. If the rules are not carefully written, the AI can attack the lead pilot, which would also destroy the pair.
[169] Recall that the first term of the Fourier Series is a constant.

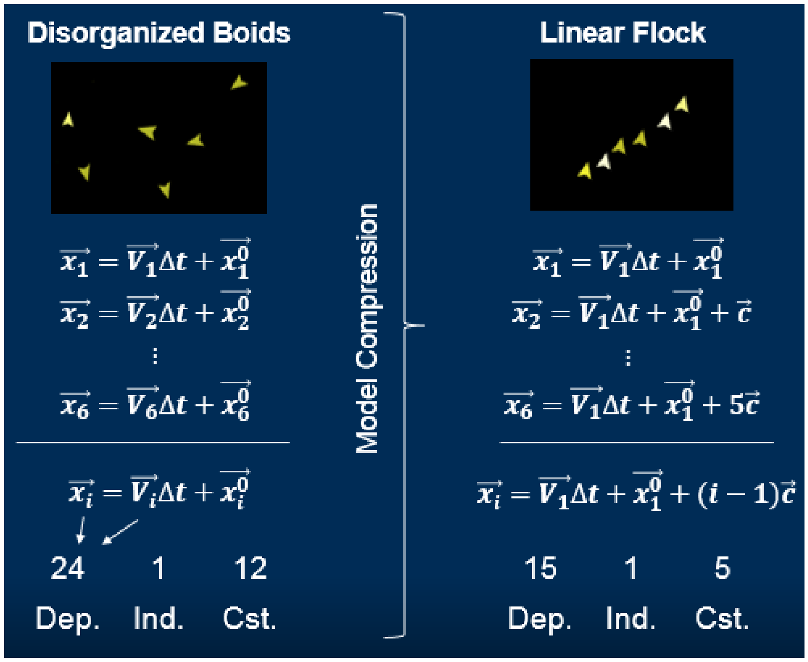


Figure 18 – Boids self-organization and model compression

Before the line had formed, the system of equations for the position of the boids was,

$$\vec{x_i} = \vec{V_i}t + \vec{x_i^0} \tag{1}$$

The i[th] boid possessed its own velocity vector, $\vec{V_i}$, and its own initial position, $\vec{x_i^0}$. After the line formed and stabilized (assuming the boids are spaced equidistantly), every boid adopts the same velocity (speed and heading) as the lead boid. Therefore, assuming the distance between the given boid and the lead boid increases with index number, the system becomes,

$$\vec{x_i} = \vec{V_1}t + \vec{x_1^0} + (i-1)\vec{c} \tag{2}$$

where $\vec{c}$ is constant the relative displacement among boids.[170] The other boid velocities and initial positions can be replaced with one velocity for all boids, and one displacement, which then simplifies much further. The original system had 37 variables: the two-dimensional positions and velocities of 6 boids (a total of 24 dependent variables), time (1 independent variable) and the initial positions of the boids (12 constants). The linear flock system only has 21 variables: a single velocity (2 dependent variables), six positions (12 dependent variables), an index variable (treated as an independent variable, or 5 non-zero constants), time (1 independent variable), and the inter-boid spacing (2 constants). In this simplistic example, the velocities of the other boids are no longer necessary. In terms of space complexity, the number of variables in the system has decreased by,

$$24 - 14 = 10 \rightarrow \text{change in space complexity}$$

The qualities that made these properties 'interesting' from an ABM standpoint (that they would obey their own internal rules) appear to have been usurped by the formation of this spatial pattern.[171] Note also that if there were more boids in this line, the amount of data compressed would increase. Here, an argument can be made that for the linear flock abstraction to be meaningful, the properties of the largest possible flock must be the same as the properties of the smallest possible flock (i.e. a long line should have the same

[170] The line depicted in Figure 18 shows a "wobbly line." In that case, although the relative distance between each pair of birds is constant, there are multiple relative distances, which would increase the number of constants in the equation.

[171] Again: in general, *any* dependent variables (y_i,y_k) that can be represented as a metric space can be used to detect self-organization. The criterion is that there exists a periodic function, f, such that $y_i = f(y_k, x)$, where x is one (or more) independent variables in any metric space, not just time. Whether that pattern is useful depends on the context.

properties as a short line; no dependence on the number of boids). Although this possibility will be discussed in the results, it will not be extensively studied in this thesis.

Although the flocking model is used solely for its self-organization, a Master's thesis by Jason Smith found that the behavior of very large groups of boids[172] (and real animals) can exhibit fluid-like behavior, and thus, can be modeled using a short set of equations with few parameters [210]. Not only is this an excellent example of data compression, but the fact that the low-level boids model can be replaced by a fluid-motion model indicates that the boids model can produce emergent behavior in the limit of very large flocks. Smith attributes the fluid-like motion to the separation and cohesion rules of the boids model, not emergent properties of small-to-intermediate sized flocks. This suggests at least two possibilities: (1) small flocks do not have emergent behaviors, (2) small flocks do have emergent behaviors, but those behaviors somehow become irrelevant as small flocks are subsumed into massive flocks. If the second is correct, then it may be that emergent properties do not necessarily act as a kind of ladder from low, to intermediate, and then to high levels of abstraction. Rather, the emergence that appears at small scales may come undone or simply have no effect on large scale behavior.[173] On the other hand, it may be that such a ladder does not exist when the large scale is fluid-like, but does exist if the large scale is solid-like.[174] Studying this sort of multi-level scaling phenomenon is a topic for future study.

---

[172] Smith references the original boids model by Reynolds, but does not utilize the NetLogo implementation.
[173] In this sense, imagine a functional decomposition where some intermediate-level functions have no upward path to higher-level functions. They simply dead-end.
[174] It is well known that small dust particles or surface defects act as nuclei for crystallization. Perhaps certain organizational structures are more "rigid" than others, and the extent to which this rigidity continues into larger scales determines the effect that small-scale emergence has on larger scales. For example, grains of

3.4 Synthesis of Key Observations: Measuring versus Modeling Emergence

Although the criteria for emergence will be given in CHAPTER 4, the preceding arguments are enough to clarify precisely *why* the various measures of complexity, emergence, and now self-organization have all been rejected: this thesis is based on the premise that emergence cannot be measured, it can only be represented.[175] Simulations and experiments reveal the myriad interactions that ultimately culminate in the high-level, collective behavior that this thesis refers to as emergent behavior. Therefore, simulations can be used to depict / illustrate / model[176] / represent emergent behavior. Emergence, itself, is not a measurable quantity. Rather, it is the byproduct of the organization / structure of a collection of components, and is only observable when two collective objects interact. Furthermore, it is not until after the interaction is observed that one can hope to write an equation characterizing the behavior of the collective objects. For example, it is not possible to know from the properties of atoms alone that a helical strand of DNA can replicate itself, except by an extraordinary stroke of luck or genius. One would have to observe it in its environment to watch the process unfold. Referring back to the example of Epstein's bees, it is impossible to predict how a real bee will respond to a real wasp without the wasp. Regarding the various measures of complexity, emergence, and self-organization, it may be that a mathematical formula using degrees of freedom, information entropy, and/or Kolmogorov complexity will one day prove to be a better basis for the

---

sand are solid, but the grains have no cohesion, and so they behave like fluids at large scales (depending on the applied stress). One immediately wonders how this would generalize to fractals.

[175] This is an interpretation based the numerous aforementioned articles stating that emergence can only be observed by running the simulation / conducting the experiment (the first time). Once it has been observed, this thesis holds that it can be predicted.

[176] Here "model" is used in both the general sense and the strict "mathematical equation" sense discussed in Chapter 2. A simulation is nothing more than the numerical integration of an unsteady partial differential equation (or the equivalent solution mechanism for discrete / quantized data).

change in the measured complexity of the system,[177] but first someone must develop an unambiguous connection between the two. This thesis takes a much simpler approach to associate self-organization with the number of emergent properties attributable to a collection of components.

Ultimately, the issue underlying Research Question 1 is the following: when does a *self-organized pattern of birds*, for example, become a *flock* with properties and behaviors of its own? Borrowing a dichotomy from Winning and Betchel, how can the emergence of a pattern be distinguished from the emergence of a being [211] *within the context of a model*?[178] Crutchfield provides a clue: "moving from the initial intuitive definition of emergence to the more concrete notion of pattern formation and ending with intrinsic emergence, it became clear that the essential novelty involved had to be referred to some valuating entity," [199] arguing then that, "the observer is that which recognizes the 'something' … [and] is one that has the processing capability with which to take advantage of the emergent patterns" [199].[179] That is not to say that a self-aware entity that passes the Turing test must "observe" the emergence, but simply that some other entity must be present that can capitalize on the existence of this new property. This view is consistent with a body of literature summarized by Bonabeau:

> *Most of the definitions of emergence related to the idea of levels rely on the existence of an observer or of some device capable of observation… but for some others, a definition of emergence must not include any reference to a cognitive observer, i.e. no mental states must be involved in the definition. This is not necessarily in contradiction with the use of "observational mechanisms", but such mechanisms must not be taken … because of the 'structural plasticity of biological systems.'* [200]

[177] Assuming the premise of this thesis holds, there is a connection analogous to space complexity. Otherwise, there would have to be some other connection between measurements of organization and the discrete number of new emergent properties unrelated to space complexity.
[178] This caveat, along with the idealizations in 2.2.2, provides the ontological grounding for all objects discussed in this thesis.
[179] Note that Crutchfield uses the term model in a sense more general than this thesis.

Without that last piece of information (i.e. the presence of multiple systems that can somehow affect and be affected by emergent properties) it will be mathematically impossible to prove that emergence has actually taken place. That is the subject of CHAPTER 4.

CHAPTER 4. COMPLEXITY AND EMERGENCE DEFINED

Speaking on causation across/among levels of abstraction, Mitchell argues, “It may well be that the complete causal process is enacted by *physical* entities; what else could there be? But at the same time there will not be a representation that completely captures this process in terms of *physics* entities” [144]. Therefore, studying emergence requires new ways of thinking about and describing familiar objects in addition to postulating new ones. Borrowing an example from Abbott [72], triangles are not physical entities (although physical objects can form triangles), and their properties (e.g. the sum of their interior angles equals π) are derived from the axioms of mathematics, not the study of physics. Therefore, a property such as the magnetic dipole of the water molecule is caused by the atoms *and* the triangular shape of the molecule. It is easy to take this for granted, but doing so creates problematic ontologies that stifle the study of complexity and emergent behaviors. Writing on the subject of new ways of thinking, the Complexity Primer makes one particularly remarkable suggestion to those studying complexity, “Combine courage with humility. It takes courage to relinquish control, encourage variety, and explore unmapped territory. It takes humility to accept irreducible uncertainty, to be skeptical of existing knowledge, and to be open to learning from failure” [70]. With that said, here goes something.

As the Research Objective of this thesis (Section 1.7) states, the goal of this thesis is to combine three basic steps:

1. To develop a method for making non-decomposable, quantifiable properties and behaviors traceable

2. To associate specific non-decomposable, quantifiable SoS properties with specific system-level patterns of interaction
3. To demonstrate that one or more patterns of interaction are exploitable

The term "exploitable" can mean many things to many people. As discussed in Section 2.2.2, it is impossible to fully simulate the environment that a system operates in, which not only eliminates sources of instability and complexity, but also adds sources of instability and aleatory uncertainty unique to the computer executing the simulation, and the program being executed. In order to restrict the scope of this thesis to simulated SoS while striking a balance with the needs of Fleet CBAs or Fleet Synthesis studies, ***exploitable*** can be defined as "the ability to be affected by a simulated entity such that one or more of the simulation idealizations (the simplicity, indivisibility, predictability, or persistence) of a SoS are undermined." Once again (from 2.2.2), a simulated system is not guaranteed to inherit any idealization from its simulated components. However, if the pattern underlying a SoS property definition is stable, then it is also predictable, persistent, simple (in the sense that it is known), and indivisible for the duration of the pattern (in the sense that the pattern is indivisible). To irrevocably disrupt that pattern is to exploit the property, and hence, the SoS.

To associate a specific SoS property with a specific system-level pattern of interaction is equivalent to associating that property with a specific entity (the pattern). This is a stronger form of traceability, and it is the form of traceability desired in this thesis. Since the SoS property is quantifiable, the essence of this stronger traceability is to express that property as a function of metrics associated with that pattern. Otherwise, as Kitto warns [73], an infinite list of arbitrary properties could be generated. In this way, the property of the SoS is meaningfully traceable to the collection of components engaged in the pattern, without being defined in terms of the properties or behaviors of any single component (i.e. without becoming decomposable). However, since a pattern is generated by a set of objects, rather than an individual object, the formation of the pattern must somehow correspond to a quantity that, when measured, can indicate the onset of this association so that traceability (in the general sense) can be unambiguously determined. This thesis will rely on the information compression characteristic of patterns to take this measurement.[180]

Since simulated entities exist and have properties by definition, their properties can be arbitrary and meaningless. For a simulated, traceable property within a simulation to have application in real systems, it must possess a quality that makes it consequential (i.e. that demonstrates it causes something to occur). Without this, the only conclusion one could draw from the observation of self-organization would be that a pattern exists. Furthermore, it would be impossible to empirically validate since inert properties cannot be sensed/detected. Within the context of a model, to prove that a simulated property is consequential, it must interact with the property or behavior of some other object. Therefore, in order for a self-organized pattern to be called a SoS (an entity unto itself), it must be shown that the SoS interacts with another entity (a component, system, or SoS apart from itself). In other words, "I interact therefore I am."[181] As the equation-free modeling approach demonstrates, it is possible to identify such properties without

[180] In other words, pattern recognition will form the logical justification for a declaration of traceability (something exists and is traceable). Other authors might adapt entropy or some other metric for this purpose. However, the actual act of tracing one property to another will be take the form of fitting a function to a set of data.
[181] Aphorism credit: James Pagan. See also reference in Footnote 71.

enforcing traceability. However, in this thesis, the process of searching for SoS-entity interactions is also the process of establishing traceability.

4.1 A Pragmatic Definition of Emergence

Recall Research Question 1, “Which essential features of emergent behavior constitute necessary conditions that can be implemented in a mathematical/computational model?” In the absolute broadest sense, the ability to attribute a behavior to a (structurally) higher-level entity is a N.C. [68]. The key is to identify additional constraints that justify this attribution. While it is premature to call this list sufficient, the subject of emergence has been bounded enough to permit the presentation of formal hypotheses. Therefore, from the literature, the **Answer to Research Question 1** is the following list of necessary conditions (referred to as ***NC-RQ1*** going forward):

1. Dynamical
2. Decentralized
3. Structurally Decomposable
4. Self-organization of Components
    a. Nonlinearity of the model
    b. Variability in Component Arrangement
    c. Periodic Behavior
    d. Model Compressibility / Change in description length
5. Objective Unpredictability
6. Not Functionally Decomposable
    a. Not Aggregative

b. Irreducibility

c. Functional decomposition contains cycles; layered graph / hypergraph

7. Attributable to a Higher-Level Object

Once a system has self-organized, a pattern is present, which implies that the model can be partially compressed in space (change in description length / compressibility). In order to study the behavior of the self-organized object without loss of information, it is still necessary to run the full simulation since, at any point, an instability can disrupt and destroy the pattern. Nevertheless, so long as the pattern is stable, the self-organized object can be treated as a single entity, thereby reducing the number of distinct objects in the simulation. If the pattern is stable enough, the original nonlinear model for the components can, in principle, be replaced with a model of the self-organized object. That is, after the emergence has been observed in a simulation or experiment, it is no longer "surprising" but remains objectively unpredictable because in order to observe it again, the simulation or experiment must be repeated exactly as it was before. For example, human consciousness is not surprising, but it is impossible to recreate the same person. Finally, continuing with the example of human consciousness, since it is impossible (given our current understanding of the universe) to attribute consciousness to atoms, emergence implies positing the existence of a human (object at a higher level of abstraction than its components[182]) to whom conscious thoughts can be assigned as a property. In this sense, the concept of emergence used in this thesis is consistent with every-day ontology and language where objects such as "dog," "sky," and "cloud" can be readily identified and

[182] Again, the "levels" of abstraction need not be hierarchical. Any categorization could do, including heterarchical structures.

mathematically modeled (although there is always room for debating the finer distinctions between, say, breeds of dog, or the canine evolutionary tree).

Combining these conditions with the definitions provided in Section 2.3, and the above discussion, the following pragmatic definition of emergent property is proposed,

<u>Pragmatic Definition of Emergent Property</u>: - If a (simulated) SoS can be shown to interact with another (simulated) entity using its SoS-level properties subject to NC-RQ1, then that property is an emergent property.

Emergent behavior[183] merely follows from the definition of behavior in Section 2.3. Recall here that SoS is defined using the model-based definition provided in Section 1.5, and so the definition naturally extends to systems and their components.

With these definitions and conditions, it is now possible to investigate the practical steps required to quantitatively associate emergent properties with self-organization. Since the experiments in this thesis are numerical it becomes necessary to ask,

<u>Research Question 2</u>: What minimum set of data is necessary to simulate a SoS that satisfies the requirements for emergent behavior?

4.2 Requirements for Simulating Emergence in SoS

As discussed in Section 2.2.2, the only simulated objects that are ontologically grounded are the basic components in a simulation (the ones deliberately coded by a

[183] This chapter concludes the argument that weak and functional emergence are the only categories needed for complex systems behavior prediction, and outlines the numerical approach for doing so. This bears some resemblance to structural functionalism in sociology. Functionalism, as a term, has related uses in biology and philosophy. This thesis will not explore the literature on functionalism in any further detail. The interested reader is referred to [126] [343] [344] [345] for more information.

programmer). They exist by definition and can possess the qualities of being indivisible, persistent, simple, and predictable. Therefore, in order for a system to be simulated in an ontologically useful sense, there must be some circumstance in which it can be said to possess those same idealizations while also satisfying NC-RQ1. In order to possess indivisibility, components must self-organize and exhibit a bounded pattern of spatial arrangement that is robust to perturbation.[184] That is, the *arrangement* is indivisible despite being structurally decomposable, up to some maximum perturbation. Going forward the arrangement of components will be referred to as the system, as is common practice. In order to possess persistence, the component-level interactions that prompted the self-organization must persist or develop into some form of equilibrium interaction. For example, social self-organization might have been prompted by a sudden increase in communication between individual people, which after some time settles into a regular pattern of communication (a relationship). Since the *component-level interactions* are persistent and regular, the system persists.

In order to be simple, the system must possess one or more distinct properties that other entities can interact with for as long as it persists. In this way, the system also becomes predictable, in that it can be the cause of an effect, and it can be affected in a traceable way. As a result, it becomes possible to derive *interaction rules* for the system.[185] Therefore, in the context of bottom-up simulations, the qualities of simplicity and

[184] Although this can have analogies in other coordinate systems and spaces, such analogies will not be explored in this thesis.
[185] Whether the rules take the form of a math equation or computer code is irrelevant.

predictability are coupled.[186] The meaning of the term distinct will be formalized in the upcoming sections.

What is not necessarily clear from the simplicity and predictability of a simulated SoS is the way in which the underlying systems will be affected by the higher level behavior, unless, of course, the pattern equations are somehow invertible. When combining simplicity and indivisibility, consider that a SoS may change its underlying structural composition while maintaining the same property. In this case, there is some ambiguity that can be clarified within the context of the problem being studied.[187] For example, a human being possesses consciousness before, during, and after drinking coffee, even though the behavior of his/her endocrine system changes in response. There is no reason to argue that the definition of consciousness is somehow undermined by drinking coffee. See Abbott's discussion of reification machines and levels of abstraction for more information [77] [212]. An additional issue with SoS is the number of environments the SoS operates in. For example, sentient life can be said to operate in a physical, external environment, as well as a psychological, internal environment. This possibility may be explored in future work.

With this in mind, the **Answer to Research Question 2** is that a simulated SoS is properly defined when the following four conditions are met (going forward, ***NC-RQ2***),

1. Its constitutive components are engaged in self-organization.

---

[186] They may be decoupled at the component level, but cannot be at the SoS level.

[187] Although it is outside the scope of this work, this author considers this ambiguity appropriate for open systems.

2. The conditions for its dissolution are defined such that a SoS can be simulated independently of its constitutive parts (in principle, see Section 3.2).
3. Its quantitative properties and their equations are identified.
4. Its interactions and their equations are identified.

Condition 2 is essentially the emergent behavior equivalent of determining the bounds in which a model is valid. The self-organization and dissolution criteria would essentially serve as underlying assumptions for a higher level simulation (wherein the SoS is implemented as a component). Since this thesis will not undertake the effort of creating a set of multi-scale/level compatible simulations, the second requirement of the above list will be enforced by visual inspection of the data. Furthermore, since multi-scale simulation is outside the scope of this thesis, there is no need here to convert interaction equations into a higher-level interaction rule set.

Since the steps for Condition 1 and 2 are those of pattern recognition, this thesis will utilize existing tools and techniques, which are specifically identified in Section 3.1 and 5.2. Conditions 3 and 4, however, corresponds to gaps in the literature, which leads to Research Question 3:

Research Question 3: How many nontrivial quantitative emergent properties must a simulated SoS have?

Research Question 3 is analogous to asking whether or not a simulated SoS has any properties at all. If the answer is no (zero properties), then either it is merely a pattern, or the simulation does not possess enough information to characterize the emergent behavior.

Then, once a number of possible properties has been determined, Research Question 4 follows:

<u>Research Question 4</u>: Which quantitative properties are candidate emergent properties of a simulated SoS?

To answer Research Question 4 there must first be some method for enumerating candidate properties, and then an approach for confirming that a separate object in the simulation can interact with that property (subject to Ockham's Razor).[188]

4.3 Traceability and Associating Self-Organization with Emergent Properties

In keeping with the unpredictability of emergent behavior, it is impossible to determine which properties will appear without knowing (1) how the environment, other agents, or other SoSs will sense, respond to, or interact with them and (2) how the SoS itself will capitalize on the novel property to sense, respond to, or interact with other entities. In this sense, the exact numbers of behaviors that will appear is context dependent (see discussion on environment idealization and bee hives in Section 2.2.2-2.2.3). If taken to refer to this context dependence, Research Question 3 is unanswerable.

On the other hand, once self-organization has occurred, the resulting data compression that results from the formation of the pattern is intrinsic to the system. Therefore, it may be that the ultimate capacity of a system to exhibit any manner of emergent property is contingent on the amount of data that it compresses. In this case, it is possible that the upper bound on the number of emergent properties it can have is a function

[188] Kim's causal exclusion argument would apply here except that this thesis takes a different position from Kim: the fact that two causal models are redundant does not make them mutually exclusive.

of this data compression. In this sense, an **<u>Answer to Research Question 3</u>** is proposed in the form of Hypothesis 1.

4.3.1 Hypothesis 1: Upper Bound on the Number of Properties

As discussed in CHAPTER 3, self-organization can be detected by fitting a Fourier Series to a sequence of relative property or relative behavior data generated by two or more components (subject to restrictions). Recall that data compression in a model means a reduction in the number of dependent variables. The equations and constants required to fit the pattern equations are not incorporated into this hypothesis, but they will be discussed in the analysis of the results.

<u>Hypothesis 1</u>: *If two or more systems exhibit self-organization (exhibiting pattern R) such that the space complexity, C_S, of the pattern model, M_R, is lower than the space complexity of the model that produced that pattern, M_0, then data compression can be achieved and the maximum number of emergent properties, N_{max}, that can be assigned to the SoS made up of those systems is equal to the number of variables eliminated by that data compression plus one:*

$$C_S(M_R) < C_S(M_0) \rightarrow N_{max} = C_S(M_0) - C_S(M_R) + 1$$

Hypothesis 1 suggests, above all else, that the number of emergent properties an object can have is finite. To suppose that this upper bound is a linear function of data compression may be naïve, but in the absence of other information, nonlinearity would introduce too many confounding factors to test, and merely stating that an upper bound exists cannot be empirically falsified. Furthermore, Hypothesis 1 is strict in the sense that it argues the upper bound is independent of context![189] Finally, the +1 in Hypothesis 1 is an acknowledgment

[189] A question not addressed by Hypothesis 1 is whether the upper bound on the number of properties is "for all contexts" or "for each contexts." For example, a collection of atoms forming a material can have an

that every set automatically inherits "the number of elements in the set" as a property.[190] Using the example in Section 3.3, the flock of six boids can have up to 10 emergent properties.

Hypothesis 1 possesses an antecedent that is not controversial and can be verified using well-established tools. The consequent of Hypothesis 1 is falsifiable in one of four ways: The strongest way is to show that a SoS possesses more viable candidate properties that permitted (see Hypothesis 2), and this is the approach that will be used in this thesis. A second way is to fail to generate any viable candidate properties for any SoS considered in this thesis (e.g. even something so ubiquitous as a centroid). A third way (albeit weaker and arguable) is to demonstrate that the number of fitting coefficients and constants exceeds the number of variables compressed. A fourth way is to demonstrate that fitting parameters are not constant and also exceed the number of variables compressed.

Supposing, then, that the number of properties is finite, the next challenge becomes the matter of identifying those properties (i.e. actually tracing cause and effect among properties). An **Answer to Research Question 4** is proposed in the form of a final set of criteria, which, due to the coupling between property existence and interaction identification, contains nuanced definitions which make the resulting hypothesis weaker or stronger.

4.3.2 Hypothesis 2: Property Identification and Association

Hypothesis 2 contains terms that must be carefully defined in order to be falsifiable. They are: ***distinct*** and ***interact***. The definition of interaction relies on yet another term:

---

electrical property, thermal property, and mechanical property. Hypothesis 1 does not distinguish between having up to Nmax "electrical properties" or Nmax properties overall. Currently, this author is leaning toward "for each context" where the context is defined as the environment of the SoS.

[190] Whether the number of elements in the set qualifies as an emergent property is subject to Hypothesis 2.

associate. Two definitions are given for distinct, and two qualifications are given for associate, resulting in four possible combinations of definitions. In order to avoid confusion, please note that since Hypothesis 2 can take on four different meanings, there are four possible ways to falsify Hypothesis 2.

The claim that properties are distinct can rapidly become nuanced because some properties lose their utility at higher levels of abstraction. For example, the center of gravity is meaningful for a ship, less so for a strike group, and useless for the entire US navy. Even a property as ubiquitous as mass might not be as useful as density when, for example, a system is open. Definition 2a impose clear restrictions on the meaning of the word distinct.

Although *interact* has been defined in Section 2.3, much of the literature on reductionism (particularly Kim's causal exclusion argument [213]) centers on the observation that the low-level components of one SoS are necessarily interacting with the low-level components of a different SoS whenever two SoSs are directly interacting. Therefore, this thesis will err on the side of attributing causation to the interaction that is most closely associated with the relevant dynamics. This is clarified using Definition 2b.

<u>Criteria for Identification of Emergent Behavior from Numerical Data:</u>

If a self-organized set of systems called SoS possesses a set of properties or behaviors, P(SoS), such that one or more properties or behaviors, P(SoS), are <u>distinct</u> from those of its constitutive systems, and SoS <u>directly interacts</u> with another system or SoS using elements of P(SoS)*, then SoS satisfies NC-RQ1 and NC-RQ2 and by pragmatic definition, P(SoS)* are emergent properties or behaviors.*

In order for this set of criteria to be tested, it must be shown that the conditions presented in this claim consistently identify weak emergence and functional emergence in the data-set of a group of interacting components (those terms were defined in Sections 1.7 and 3.2). This leads to Hypothesis 2:

Hypothesis 2: *The Criteria for Identification of Emergent Behavior from Numerical Data are sufficient for the identification of weak and functional emergent properties in complex systems and Systems of Systems.*

In other words, if the models involving higher level properties are consistently more accurate and simpler than those involving low level properties, then an emergent behavior is taking place even if the regressions derived to model these interactions are only approximate. Note that this argument is compatible with the argument by El-Hani and Pihlström that "an emergent property can be regarded as real to the extent that it provides a more efficient description (for some purpose) of the configurational pattern with which it is identified than a micro-level description of that same configuration" [204], as well as the argument by Dennett [194] which they also cite. However, this hypothesis goes further than El-Hani and Pihlström. Not only does it require a more efficient description, but the description in question must represent a measurable interaction between two entities (at least one of which is self-organized).

It is worth briefly remarking on a key term in El-Hani and Pihlström's argument: "an emergent property can be regarded as *real*" (emphasis added) [204]. Most of the scientific literature does not concern itself with questions of whether something is real or not (quarks are real, as are atoms, molecules, fleets of ships, human beings, solar systems, etc.). The problem is not absent from the literature, of course, but it takes on the form of statements like "correlation is not causation" or terminology like "spurious regressions." Engineers live in a world of models, and the question to an engineer is: can the model be validated? How well does the model conform to reality? To a modern engineer, engaging with reality is a matter of conducting an experiment and measuring a quantity. It is not a matter of determining whether or not atoms truly exist in a fundamental sense. The ambiguity that philosophers confront regularly in their literature is pushed onto the empiricist to resolve, as though any question about reality can be answered with an adequately constructed experiment (sometimes this expectation is simplistic [156]). Unfortunately, philosophers do not have that luxury since the utility of experimentation itself can be called into question. The subtle difference between the questions asked by philosophers and the questions asked by engineers is probably a significant contributor to the confusion regarding complexity, emergence, and complex systems in the literature of the various disciplines (not to mention the intersection between those ideas and number theory / information theory as seen in this thesis). This is likely the reason that the clever machine learning approach by Kokar et al. [97] failed to produce a coherent definition of emergent behavior (although their study still yielded several useful results and discussions). The various disciplines attempt to discuss similar concepts using similar, or even redundant, terminology but they do so with different priorities and assumptions in mind, which can easily mislead a reader. A clear and consistent ontology is needed in order to study emergence. Systems engineers cannot avoid the issues tackled by philosophers. This thesis, as suggested earlier, takes the stance that everything is a model, and thus the objects described by those models and the behaviors predicted by those models are real. It also determines that the forms of emergence relevant to SE are weak emergence and functional emergence. This makes questions of emergence accessible to engineering methods as well as the scientific method because in most practical cases, whenever a scientist makes a prediction about the pressure of a fluid, or the position of an atom, those predictions start and stop with the model used to make the prediction. The purpose of this chapter is to render definitions specific, and to relate those definitions to measurable quantities.

Hypothesis 2 is strict (sufficient condition) in order to make it falsifiable by numerical experiment. If the hypothesis were written to say that the criteria are necessary conditions, rather than sufficient, the experimental methods in this thesis would not be able to distinguish between a deception and true emergent behavior. A mathematically rigorous approach to distinguishing between a deception and an emergent behavior is outside the scope of this thesis (there is no alternative method to compare to). The methods used in this thesis for mining the simulation data and generating a set of candidate properties are discussed in CHAPTER 5.

The details on how to falsify this hypothesis are depend on Definitions 2a and 2b, which are described below. Overall, and in layman's terms, this hypothesis can be falsified in two ways: show that the criteria are bad, or show that the criteria are not good. To show that the criteria are bad, it must be shown that high-level behaviors that are independent of high-level interactions satisfy the criteria. This approach tests that non-emergent and/or arbitrarily derived high-level behaviors do not satisfy the criteria for emergence. To show that the criteria are not good, it must be shown that a high-level interaction that should be emergent does not appear to be emergent, or is difficult to distinguish from noisy data. The former is unambiguous and will be the primary method for falsifying the criteria in this thesis. The latter is essentially the same challenge faced by every researcher in this field, and will be the subject of discussion in the chapters that follow. Hypothesis 1 is much easier to falsify in principle. If more properties are found than permitted by Hypothesis 1, then Hypothesis 1 is falsified. So long as Hypothesis 2 is "not bad," Hypothesis 1 will only be falsified when

At this point it is important to refer back to the Research Objective of this thesis. As stated at the beginning of this chapter, the first goal is to "develop a method for making non-decomposable, quantifiable properties and behaviors traceable," which is not the same as settling, once and for all, the debate over the definition of emergence. However, in order to be practical, the non-decomposable properties identified in this thesis cannot be arbitrary, and their attribution to a SoS must be logically grounded. The pragmatic definition, necessary conditions, and sufficient conditions listed here cannot been proven to be sufficient conditions of emergence beyond any doubt over the course of this single thesis. Nevertheless, they scope the discussion sufficiently to perform the tasks of tracing properties to collections of self-organized components, and provide useful guidelines for determining that those properties are meaningful within the context of systems engineering simulations.

The next issue is the definition of what constitutes a distinct property within the context of a simulation. Referring again to a block of ice: at the engineering level, the mass of a block of ice is understood to be the mean value of a statistical distribution since it is an open system. On the other hand, the mass of a water molecule is generally assumed to be constant. Clearly these are both conventions (real oxygen has isotopes; the mass of a real block of ice is estimated while, in a simulation, the number of molecules in a block of ice can be counted exactly). When assigning a property to a SoS, the convention used for the underlying systems may not extend to the SoS. This thesis will not exhaust the list of all possible conventions. Instead, a property will be treated as the totality of the mathematical expression used to define it for the purposes of the simulation. In this way, equations can be directly compared and visually distinguished. Recall that an emergent property can be expressed as a linear function because the basis of self-organization in NC-RQ1 is nonlinearity, which is intrinsic to the process of self-organization.

4.3.3 Definition 2a: Property Distinctiveness

Definition 2a - Property Distinctiveness: *Let P(SoS) be the set of all properties and behaviors possessed by a self-organized set of n systems called SoS. Let P(A) be the set of all properties and behaviors possessed by a single system, A, within SoS* ($A \in SoS$).

(2a-i) Relaxed: *The set of properties possessed by SoS cannot be identical to the set of properties possessed by any one constitutive agent:*

$$P(SoS) \cap \left(\bigcup_{j=1}^{n} P(A_j) \right) \neq P(A_k) \; \forall \; A_k \in SoS, k \in [1, n]$$

(2a-ii) Strict: *The set of properties possessed by SoS cannot contain any properties possessed by its constitutive agents (i.e. P(SoS) = P(SoS)*):*

$$P(SoS) \cap \left(\bigcup_{j=1}^{n} P(A_j) \right) = \{\emptyset\}$$

In other words, the strict version of Definition 2a (2a-ii) means that a SoS cannot directly inherit any property from its constituent systems whatsoever (e.g. the “position” of a SoS cannot be a copy of the position of any one constituent system). The relaxed form of

Definition 2a (2a-i) permits the SoS to inherit some properties directly from its constituent systems so long as not every property is directly inherited, and so long as the properties of one system are not all inherited. This is the extent to which property distinctiveness across levels of abstraction will be enforced in this thesis. Definition 2a essentially acts as a kind of necessary condition to filter sets of candidate properties (the set size being restricted by Hypothesis 1).

To facilitate the search, however, it is reasonable to incorporate structural information provided by well-known metrics in the search for new properties. Minati provides a convenient list of properties that a SoS could potentially acquire from its constituent systems without directly inheriting properties from an individual constituent:

> *"(1) Properties of the values acquired by mesoscopic variables…such as any regularities including periodicity, quasi-periodicity, chaotic regularities possibly with attractors …*
>
> *(2) Possible statistical properties of sets of meta-elements detected by suitable techniques …*
>
> *(3) Properties, e.g., geometrical and statistical, of sets of generic agents constituting mesoscopic variables;*
>
> *(4) Properties related to the usage of degrees of freedom …*
>
> *(5) Relationships between properties of sets of clustered generic agents and, macroscopic properties such as density, distribution … percentages;*
>
> *(6) Properties of the thresholds adopted for specifying the mesoscopic general vector;*
>
> *(7) Levels of ergodicity or quasi-ergodicity;*
>
> *…*
>
> *(9) Possible topological properties of network representations such as power laws and scale-freeness."* [191]

Geometric properties, of course, include shape parameters (characteristic lengths, angles, numbers of sides, etc.). Values acquired from periodic functions (such as the one fit to the self-organization pattern) include the period and amplitude of the oscillation (i.e. one SoS property may be the amplitude of oscillation). Regarding, network properties, Green and Newth go even further than Minati, arguing that "virtually any complex system inherits properties of graphs" [76]. If Green and Newth's statement is interpreted with its full force, it could prompt a near automatic violation of Hypothesis 1. It does not, however, because according to Hypothesis 2, the property is only emergent if it participates in an interaction.

What remains ambiguous in the wording of Definition 2a is whether or not certain statistics of the set of systems should be considered "the same" as the property of a system in that set. For example, suppose all properties have the same units, and the property of the SoS is the maximum value of all system properties. Clearly, at every instant in time, $\max\left(\bigcup_{j=1}^{n} P\left(A_j\right)\right) \in P(A_k)\ for\ some\ A_k \in SoS$. This author considers the maximum statistic "distinct" from the property of a system so long as A_k can change during the simulation (i.e. the maximum is held by different constituents over time).

4.3.4 *Definition 2b: Interaction Detection and Association*

Definition 2b is motivated by the reality that if a SoS is interacting with an external object, then at least one of its constituent systems is also simultaneously interacting with that same external object (or its components). However, this thesis aims to attribute causation to the SoS, not the constituent system, otherwise Hypothesis 2 is falsified. Therefore, in order to claim that the SoS is causing the interaction, there must be a property or behavior of the SoS that is more closely associated with the change in the object's properties than a property or behavior of a constituent system. The meaning of association is clarified in subsequent definitions. Note that since interactions can be multivariate, Definition 2b seeks a set of SoS properties and behaviors that are more closely associated to the object's properties than some set pertaining to one of its systems. The added restriction is that the set of SoS properties must contain at least one *distinct* property (one property satisfying Hypothesis 2a), otherwise the SoS property set is merely a copy of its system property set.

<u>Definition 2b – Interaction due to SoS:</u> *Let $p_i = p(SoS_i) \subset P(SoS_i)$ be a subset of the set of properties or behaviors, $P(SoS_i)$, of SoS_i s.t. $p(SoS_i) \cap P(SoS_i)^* \neq \{\emptyset\}$. Let $a_i \subset P(A|A \in SoS_i)$ be a subset of the properties or behaviors of a system, A, contained in SoS_i. Let b represent a behavior of another system or SoS: $b \in P(SoS_j)\ s.t. SoS_i \cap SoS_j = \{\emptyset\}$, or $b \in P(A_j)\ s.t. A_j \notin SoS_i$. If p_i is <u>more closely associated</u> with b than a_i, then SoS_i directly interacts with A_j/SoS_j.*

Note that Definition 2b says "directly interact." It may turn out that closer associations are impossible, which would imply one of two things:

(1) Interactions were identified but were consistently of lower association strength than system-level interactions. This would imply that SoS only interact indirectly with other objects via their constituent systems rather than directly, which would suggest that higher-level models are always less ontologically grounded than lower-level models. If this is the case, it does not invalidate the work of this thesis, but it does mean that expectations regarding emergent behavior must be shifted accordingly (e.g. accepting that interactions are indirect as opposed to direct-yet-contemporaneous). By Definition 2b, this case will falsify Hypothesis 2;[191]

(2) No interaction was found of adequate strength, which means that no property satisfying NC-RQ1 or NC-RQ2 were found. Hypothesis 2b is falsified in general.[191]

[191] Strictly speaking, only a truly exhaustive test of all possible properties can completely falsify Hypothesis 2b due to its formulation. The methods used here will aim for that, but due to time constraints, this caveat must be stated.

The following definitions further clarify the meaning of the term "association." Essentially, both forms argue that a good association can be determined by combining a measure of goodness of fit with a measure of the time complexity of the model that is fit to the data. Loosely speaking, this is a combination of correlation between data sets and Ockham's razor.

(2b-i) Weak Association: *Let x, and y be independent variables. Let z be a dependent variable. Let m be some metric that measures the goodness of fit of some data regression. Let f and g be functions used for data regression. Let C_T be the model time complexity of some function. If $m\big(z, f(x)\big) > m\big(z, g(y)\big)$ and $C_T\big(f(x)\big) < C_T(g(y))$, then f(x) is more closely associated with z than g(y). If $m\big(z, f(x)\big) = m\big(z, g(y)\big)$ and $C_T\big(f(x)\big) = C_T(g(y))$, then f(x) is as closely associated with z, as g(y).*

In other words, given two different regressions on the same data set, the regression that has both a higher goodness of fit and a lower time complexity is the better regression.

(2b-ii) Strong Association: *Let x, and y be independent variables. Let z be a dependent variable. Let m be some metric that measures the goodness of fit of some regression. Let f_i be any function derived from some pre-determined set of operations. Let C_T be the (model) time complexity of some function.*

Let X_m be a set of goodness measures, $X_m = \big\{m_i\big(z, f_i(x)\big)\big\}$; Y_m similarly defined.

Let X_c be a set of time complexity measures, $X_c = \big\{C_{T,i}\big(z, f_i(x)\big)\big\}$; Y_c similarly defined.

Let X be the set of pairs, $X = \{X_m, X_C\} = \big\{m_i\big(z, f_i(x)\big), C_{T,i}\big(z, f_i(x)\big)\big\}$; Y similarly defined.

Assuming optimization objectives that maximize goodness of fit while minimizing time complexity, $\max\left(m\left(z, f(x)\right)\right)$ *and* $\min(C_T(f(x)))$, *if the Pareto Optimal subset of X strongly dominates Y, then x is more closely associated with z, than y. If the Pareto Optimal subset of X weakly dominates Y, then x is as closely associated with z, as y.*

In other words, imagine generating a set of regressions on a data set *z*, using the functions of the variable *x*. Each regression has its own goodness of fit measurement and its own time complexity. Clearly some are better than others. From that set, those that have the best trade-off between goodness of fit and complexity form the Pareto Optimal set (see [214] for more information). Generate another set of regressions using y and the independent variable, rather than x. If the Pareto Optimal set of one variable dominates the other, it is the variable more strongly associated with z. If it only weakly dominates, then they are equally well associated with z. Readers familiar with Pareto Optimal sets will recall that the true Pareto Optimal set may not be attainable; in some problems it can only be approximated [215] [216]. However, if only approximate sets are available, then the entire two-dimensional distribution of performance metrics can also be compared.

To clarify the intention and function of Hypothesis 2, Figure 19. The properties of the red flock (represented by a red line) are given by variables y_k. The properties of the blue flock (blue line) are given by variables z_i. The properties of *a single boid* contained in the red flock are given by the variables x_h. All the properties y_k can be expressed as a function the properties of the boids in the flock. This is upward causation. Clearly, the flock has a length because of the space between boids, and a position (center of gravity) because the positions of the boids can be averaged. The same is true for the blue flock. However, it may be easier and more accurate to write an equation for the time-rate-of-change of z_i in terms of y_k, $z_i = f(y_k)$, than in terms of x_h, $z_i = g(x_h)$ (the properties of *any one boid* contained in the red flock) where *f* and *g* can be any function. Therefore, it is not merely an interaction with a component, which is governed by predefined rules, but rather it is an interaction with a self-organized collection for whom no interaction rules were explicitly coded in the simulation. If this equation generalizes to all flocks of the same type, then it is an emergent behavior of that flock and it can be used to write rules for simulations of flocks.[192] This casual description will be clarified in Section 5.1.3 on model complexity calculations.

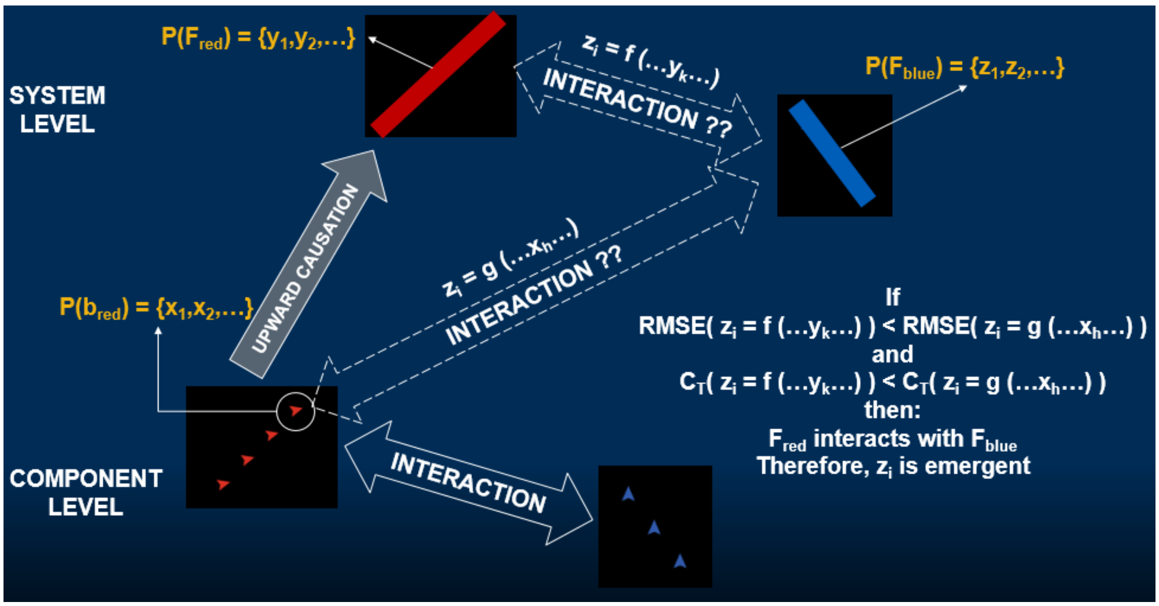


Figure 19 – Pictorial representation of Hypothesis 2 (weak association)

To clarify an aptly phrased and relevant question, the previous paragraph will be paraphrased. A question was asked: "What is the difference between an emergent behavior and the result of just running a simulation?" Clearly, every conclusion one draws from a simulation is a consequence of running the simulation. That includes emergence. However, the simulation itself is made up of pre-determined objects and their pre-determined behavior rules.[193] The simulation contains no explicit information regarding the properties

[192] The properties of the red line and blue line were denoted with variables y and z to make the argument easy to follow. Since both flocks are of the same type they must have all the same properties (y = z).
[193] Recall the response to Epstein's work in Section 2.2.3.

or behavior rules of self-organized objects. Since those properties and behavior rules belong to the higher level object, they are emergent behaviors. Those rules are nowhere written in the code or underlying mathematical model (although they are caused by that underlying mathematical model). The fact that a simple behavior rule can be written at all for the higher level object is the emergence. That rule, combined with an appropriate set of conditions for the creation and dissolution of the self-organized object can be used to create an entirely new simulation where the higher level object is coded in as the basic, idealized component (just as one codes a computational fluid dynamics solver or a molecular dynamics solver without regard for Bosons). Rather than flying boids, the simulation would be of flying lines. Those lines could then, potentially, self-organize and that new object could potentially have its own emergent properties. On the other hand, those lines could interact without ever self-organizing, in which case the hierarchy of emergent objects will have reached a dead end. Dead-ends are beyond the scope of this thesis.

4.4 Systems Science Research Q&A Summary

Recall that USN Fleet CBA and Fleet Synthesis studies demand the ability to physically and functionally decompose systems and SoS.

> **Research Problem:** *The traditional SE approaches to defining the properties and behaviors of a SoS that are distinct from those of its constituent systems lacks generality and traceability, and results in designs whose behaviors are only partially understood, the remainder of which can be exploited for some unintended purpose.*

The associated gaps in the literature are essentially caused by the reductionist methodologies that suppose complex behaviors are decomposable (they are not), which then fuels disagreement over the definition and inherent nature of SoS, complex behavior, and emergent behavior.

Research Objective: *To develop a method for rendering non-decomposable, quantifiable SoS properties and behaviors traceable to the patterns of interaction of their constitutive systems, so that exploitable patterns identified during the early stages of design can be accounted for.*

This research objective entails developing a better understanding of non-decomposable SoS behaviors (referred to as emergent behaviors on pragmatic grounds), in order to develop a method to trace them to specific constituent systems of an SoS. Although much work has been done with regards to emergent behavior, no body of work to date (to the knowledge of this author) has enabled users to unambiguously associate quantitative data with qualitative behaviors, and then assign those behaviors to higher level entities, as depicted in Figure 20.

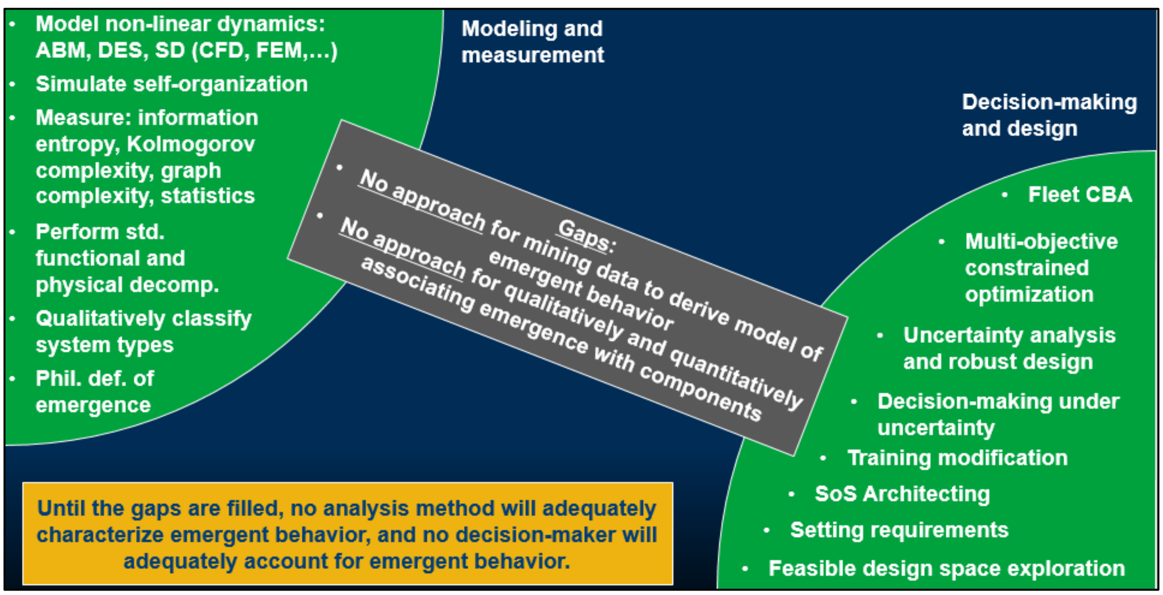


Figure 20 – Knowledge gaps in relation to existing literature

Therefore, the first research question is designed to identify the qualitative features of emergent behavior in order to guide the thesis.

Research Question 1: *Which essential features of emergent behavior constitute necessary conditions that can be implemented in a mathematical/computational model?*

The answer to this question is provided through a literature review, which culminates in the collection of necessary conditions referred to as NC-RQ1:

1. *Dynamical*
2. *Decentralized*
3. *Structurally Decomposable*
4. *Self-organization of Components*
5. *Model Compressibility*
6. *Objective Unpredictability*
7. *Not Decomposable*
8. *Attributable to a higher-level*

This thesis will conduct experiments using simulations. Within that context, a nuanced definition of SoS, and emergent behavior is provided and discussed.

Research Question 2: *What minimum set of data is necessary to simulate a SoS that satisfies the requirements for emergent behavior?*

The answer to this question is also provided through a literature review, which culminates in the collection of necessary conditions referred to as NC-RQ2.

1. *Its components are self-organized*
2. *The conditions for its dissolution are defined*
3. *Its quantitative properties and their equations are identified*
4. *Its interactions and their equations are identified*

The remaining research questions focus on practical questions of traceability. First, one must determine how many properties exist, if any.

Research Question 3: *How many nontrivial quantitative emergent properties must a simulated SoS have?*

Without first simulating or observing the SoS, the only logically defensible answer is to posit the upper bound on the number of properties a SoS can have.

Hypothesis 1: *If two or more systems exhibit self-organization (exhibiting pattern R) such that the space complexity, C_S, of the pattern model, M_R, is lower than the space complexity of the model that produced that pattern,*

M_0, then data compression can be achieved and the maximum number of emergent properties, N_{max}, that can be assigned to the SoS made up of those systems is equal to the number of variables eliminated by that data compression plus one:

$$C_S(M_R) < C_S(M_0) \rightarrow N_{max} = C_S(ABM_0) - C_S(ABM_R) + 1$$

Hypothesis 1 can be falsified in four ways: The strongest way is to show that a SoS possesses more viable candidate properties that permitted (see Hypothesis 2). A second way is to fail to generate any viable candidate properties for any SoS considered in this thesis (e.g. even something so ubiquitous as a centroid). A third way (albeit weaker) is to demonstrate that the number of fitting coefficients and constants exceeds the number of variables compressed. A fourth way is to demonstrate that fitting parameters are not constant and also exceed the number of variables compressed.

Then one must determine the criteria for assigning a property to a SoS in order to subsequently trace that property to a set of constituent systems.

> **<u>Research Question 4:</u>** *Which quantitative properties are candidate emergent properties of a simulated SoS?*

This question is answered by the following criteria and Hypothesis 2.

> **<u>Criteria for Identification of Emergent Behavior from Numerical Data:</u>** *If a self-organized set of systems called SoS possesses a set of properties or behaviors, P(SoS), such that one or more properties or behaviors, P(SoS)*, are <u>distinct</u> from those of its constitutive systems, and SoS <u>directly interacts</u> with another system or SoS using elements of P(SoS)*, then SoS satisfies NC-RQ1 and NC-RQ2 and by pragmatic definition, P(SoS)* are emergent properties or behaviors.*
>
> **<u>Hypothesis 2</u>:** *The Criteria for Identification of Emergent Behavior from Numerical Data are sufficient for the identification of weak and functional emergent properties in complex systems.*

The criteria contain nuanced definitions for certain terms (see Section 4.3). The detailed ways to falsify this hypothesis are discussed with respect to Definitions 2a and 2b, because they are coupled. In general, however, this hypothesis is falsified if no interaction can be found using properties that are distinct from those of its constituent systems, or if a decidedly non-emergent property satisfies the conditions of an emergent property.

These answers to these questions open the door to the final research question considered in this thesis, which is also the one of central concern for the execution of a CBA and FSS (see CHAPTER 5 and CHAPTER 7 for the complete development of this question and answer):

> **Research Question 5:** *Once identified, how can emergent behaviors be exploited?*

Since there is no systematic approach yet in existence, this thesis will adapt a Sensitivity Analysis (SA) that will enable decision-makers to clearly identify opportunities for emergent behavior exploitation. The adversarial boids case study and its measures of merit will be used to illustrate the sensitivity-based approach. The test for Hypothesis 3 is also a test for the efficacy of the SA:

> **Hypothesis 3:** *Targeting the system-level property will be more effective than targeting either pilot.*

Answering Research Question 5 will be the primary focus of CHAPTER 7.

CHAPTER 5. EXPERIMENT PROCEDURES AND EXPLOITATION ANALYSIS

Revisiting the Research Objective of this thesis (Section 1.7) once more, the goal of this thesis is to combine three basic steps:

1. To develop a method for making non-decomposable, quantifiable properties and behaviors traceable
2. To associate specific non-decomposable, quantifiable SoS properties with specific system-level patterns of interaction (i.e. perform the trace)
3. To demonstrate that one or more patterns of interaction are exploitable

This chapter will illustrate a method for performing each of these steps, review some tools to be used in each step, and present the experiments for testing Hypothesis 1 and 2. Following the numbering above, an analyst looking for exploitable emergent behaviors would perform the following three steps (see Table 1). That this method fills the knowledge gaps (see Figure 21) is not particularly controversial. By identifying a self-organized system, properties and behaviors become traceable to that system. By observing system-level interactions, one can infer the properties that are changing during that interaction, and so it becomes possible to associate that behavior (property change) with the system exhibiting it. Taken together, the pattern recognition and behavior association steps fill most of the gaps in Section 1.7. Finally, if the properties of the system are exploitable, then the patterns of interaction, which form the foundation of that system, are exploitable (this fills the remaining gaps).

If one wished to prove that this method cannot work there are two options. One can attempt to empirically determine that this method cannot work, which is impossible without testing infinitely many applications (though a small number of tests may suffice to show that it contains some inherent deficiency).

Table 1 – Method for bridging the emergent behavior knowledge gaps

(1) Pattern Recognition	a) Determine model / simulation space complexity b) Run Simulation c) Identify self-organization (find pattern model) An emergent behavior is now *traceable* to interacting, self-organized components, if it exists. d) Determine pattern model complexity e) Compute maximum number of emergent properties according to equation in Hypothesis 1
(2) Behavior Association	a) Continue running simulation b) Observe one or more direct interactions between components of self-organized system and components not in that system (“external”) c) Determine association between properties of external component, and emergent property of self-organized system, if any Emergent behavior has now been *traced*. Hypotheses 1 and 2 can now be tested.
(3) Exploitation Analysis	The type of *exploitability* analysis depends on the goal of the study: • Design Performance* ○ Impact on MoMs ○ Facilitate / inhibit emergent behavior • Behavior Change* ○ Rule modification (ways) ○ Design change (means) • Model Discovery * Types considered in this thesis.

As Balestrini-Robinson points out, however, the hypothesis that a method can do something or will improve something is difficult to support and to falsify [81], especially when the state of the art is largely made up of ad-hoc techniques and the literature generally lacks agreement on fundamental definitions. In the case of this thesis, there is currently no known alternative method to test against for efficacy. Second, one could attempt an impossibility proof, but as John Bell once quipped, “The only thing proved by impossibility proofs is the author's lack of imagination" [217].[194] As will be shown in this chapter, the first and last steps can be implemented using known tools and techniques. The most significant challenge in this approach is the behavior association step. Though this author does not consider the step to be impossible, there are significant hurdles to overcome in performing this step (see Section 5.3.2 in particular). Ultimately, this one step appears to be the essential gap in the emergent behavior literature.

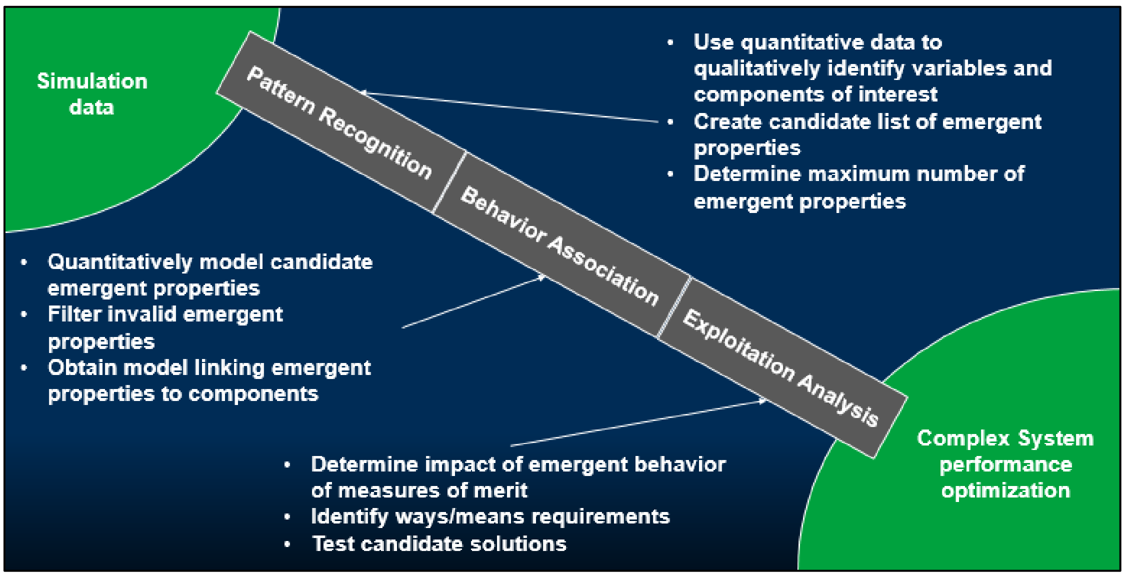


Figure 21 – Bridge across knowledge gaps (simplified Figure 20)

This chapter describes the mathematical tools and workflows needed to perform these tasks illustrated in Figure 21, and will conclude with a discussion of the case studies to be used for hypothesis testing, and the steps required to conduct the experiments. The method described in Table 1 is further illustrated in Figure 22. To implement the method, a user must obtain quantifiable data of the relevant component properties, and interactions, as well as the mission they will perform (i.e. the context in which they operate, and any purpose they may serve). If any of this data is missing, it must be supplemented by simulation or experimentation (e.g. the component properties may be known, but

[194] Though ironic, the contradiction between Bell and Gödel's theorem later in this document is superficial.

knowledge of the ways they interact may be limited), or eliminated by idealization of the component. The environment, too, must be idealized (scoped to whatever active regions are considered to be relevant). Finally, there must exist some mathematical model that can be implemented as a simulation. The assumptions used to make the model must be recorded, so that discrepancies with experimental results can be accounted for.

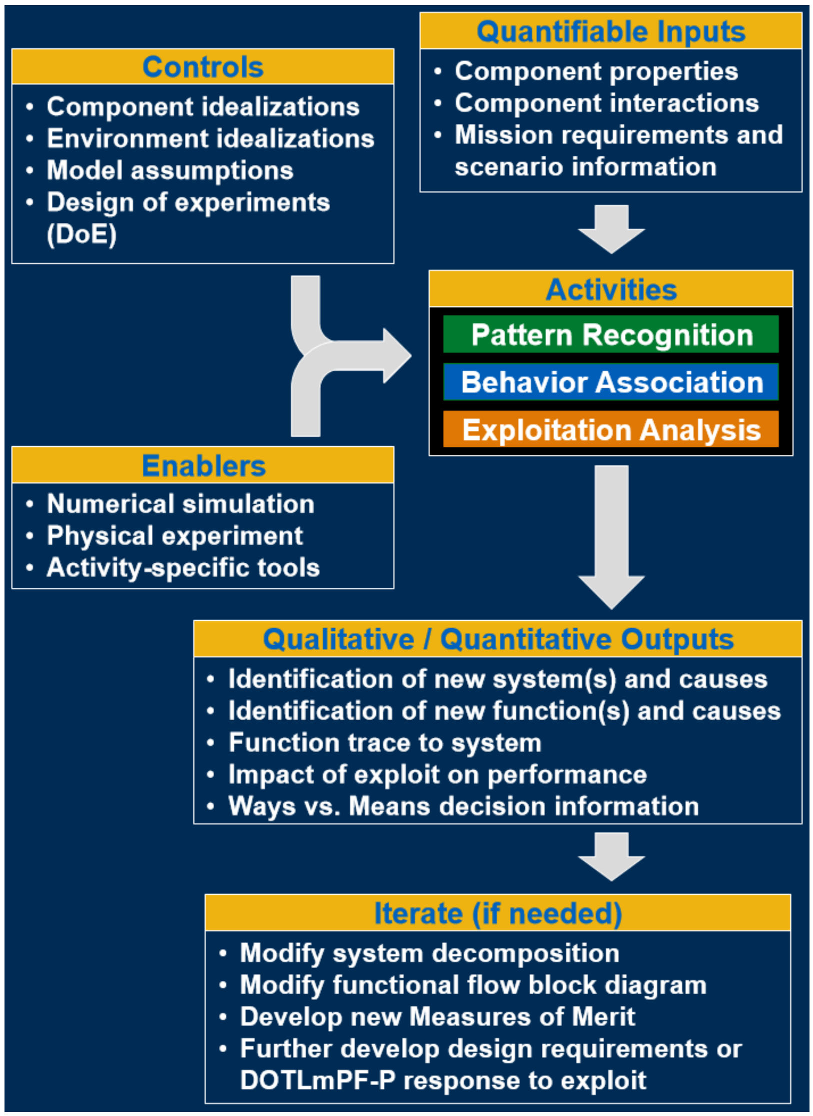


Figure 22 – Input-Process-Output diagram for emergent behavior exploitation method

As suggested at the beginning of this section, the reason for these steps, controls, and enablers is that component interactions lead to self-organization. Self-organized systems then interact. From the system-level interactions it becomes possible to identify system-level behaviors. From the system-level behaviors it becomes possible to identify exploits. From the exploits one can make ways vs. means decisions, and find that new measures of merit are needed to properly capture the performance of the exploit. The remainder of the chapter discusses how the three main steps of the method mine this data to obtain actionable decision-making / design information.

5.1 Tools for Data Mining

It is widely known that correlation is not causation, especially when it comes to emergent behavior [63]. Nevertheless, some means of detecting cause and effect is required. At least one author has suggested counting interactions directly [139]. Although that would help filter out spurious correlations, that approach does not scale efficiently with the number of properties/components, cannot distinguish between simultaneously occurring causes, and becomes challenging when interactions are persistent and continuous (e.g. many-body problems in physics). Therefore, additional tools are needed.

5.1.1 Statistical Measures

Hypothesis 2 requires techniques that measure the goodness of fit between two data sets to determine if higher-level properties and behaviors are closely related to some interaction. The naïve approach is to attempt to fit one data set to another data set over some judiciously selected time interval using standard surrogate modeling techniques, and to measure the goodness of fit (e.g. using R^2 and RMSE). While this can be implemented in order to establish a baseline, there are more sophisticated techniques for measuring the impact of one time-varying parameter on another.

Granger causality, as it is now called, is a frequently cited, widely-used statistical technique for determining whether one variable influenced the time-evolution of another variable [218]. Assuming a statistical stationary time series, the most basic form of the calculation can be expressed as:

$$\text{If } \sigma^2(X|\bar{X},\bar{Y}) < \sigma^2(X|\bar{X}) \text{ then } Y \rightarrow X \tag{1}$$

The overlined terms are time-lagged values of the time series data X and Y. The above equation tests whether the variance of the error in predicting the current value of X given time-lagged values of X and Y added together, is lower than the variance of the error in predicting the current value of X given time-lagged values of X only. If so, then Y Granger-causes X. A number of extensions to this method exist including some for nonlinear variables with applications in emergent behavior [219]. One noteworthy extension of Granger causality incorporates data-compression algorithms [220]. While that method incorporates notions used in this thesis, its compression calculations (which rely on lossless data compression algorithms) would introduce a confounding factor to this analysis, which computes model compression using an analytical approach. Therefore, it will not be included in this thesis.

O'Toole, Nallur, and Clarke[195] applied the Pearson product-moment coefficient (a very straightforward statistic) to search for correlations between properties of components in the Boids model [221]. This method is consistent with Minati's list (Section 4.3.3), as well as being easy to interpret, widely available, and also useful as a verification tool for

[195] Their definitions of weak emergence and strong emergence differ substantially from the body of literature cited throughout this thesis. Their nominal emergence is equivalent to weak emergence in this document.

analyses on the Boid model in conjunction with the results in. Furthermore, it has a natural extension to the tools that will be used for candidate property identification. However, their main hypothesis was that downward causation causes the appearance of correlations between previously unrelated variables when emergence occurs. While this has some small overlap with the hypotheses in this thesis, downward causation is not studied here.

5.1.2 Candidate Property Identification

While the above statistical methods can be used largely without modification for low-level components, an additional method must be used to generate the time-series data for candidate higher-level properties before the statistical analysis can be performed. Recently, two methods capable of efficient derivation of nonlinear models from large data sets have been published (they fall under the category of symbolic regression algorithms, which are briefly reviewed in the Appendix). Both methods take an output data set, Y, and data sets for any number of input variables, X_i, and attempt to build an analytical model relating the output to the input using a set of elementary operations, $Y = f(X_1, \ldots, X_i \ldots)$, where the result returned to the user is the functional form of f. This is a break from recent machine learning techniques such as ANN in that the goal is to provide a closed-form, human-readable equation that accurately fits highly nonlinear data. The advantage these methods have in the context of this thesis is their ability to enumerate and compute correlations for a large number of highly nonlinear combinations of variables. Doing so effectively performs an initial search of the higher-level design space (made up of nonlinear combinations of lower-level properties), filters out unlikely candidates, and returns nonlinear expressions. Therefore, while it is possible to manually generate nonlinear properties (such as the graph theoretic properties, and others suggested by Minati), these programs can create much broader lists of novel nonlinear quantities.[196]

The Sure Independent Screening and Sparsifying Operator (SISSO) [222] employs what can be thought of as a top-down approach. The user specifies which data sets to fit, which mathematical operations to utilize in the creation of the analytical expression (e.g. addition, multiplication, and simply transcendental functions), how many terms to create in the equation (these terms are called descriptors), which units of measurement each variable is expressed in, and finally, how many individual variables to include in each term. SISSO then produces the combinatorial explosion of terms that results from those arrangements of variables and operators (a 3-term equation can easily result in millions of candidate nonlinear variables). Once this master list is generated, SISSO begins estimating the correlation between those nonlinear terms and the output data. Combined with a compressed-sensing approach and efficient matrix operations to dramatically improve the efficiency (otherwise the problem is intractable), SISSO down-selects nonlinear terms to produce a list of the highest-correlated variables that satisfy the specifications of the user, and then ultimately, a human-readable set of expressions containing the top most correlated terms. SISSO currently finds application in materials science where it is used to classify higher-level material properties using properties of lower-level structures and constituent atoms [223].

On the other hand, the Sparse Identification of Nonlinear Dynamics (SINDy) [224] program takes a bottom-up approach. In the most basic application of SINDy, the approach

[196] In fact, a subset of Minati's list can be prepared in advance and provided as inputs to these programs for an even broader generation of parameters.

is based on systems whose dynamics can be modeled using a first order, linear, nonhomogeneous differential equation in time (with multivariate and potentially higher order extensions). Using the assumption of sparsity to accelerate computations, the method is capable of generating algebraic and transcendental functions beginning with one term and gradually adding more terms until an adequate nonlinear dynamics model is generated (information regarding the change in correlation with the addition/removal of each term is also available to the user). Extensions to the approach enable the reproduction of partial differential equations from empirical data sets. In a direct analogy to complex systems, SINDy has been shown to reproduce models of chaotic systems with impressive accuracy [224], and has also been used in multi-scale physics modeling to associate low-level molecular behavior with higher-level material properties [225].

While these tools perform the task of relating a high-level property to low-level properties, they do not perform the task of suggesting or deriving high-level properties from low-level data. This thesis relies on the structure of the self-organized object, and the equations derived from its organization, to provide inspiration for suggesting properties. Fortunately, in many scientific applications, the high-level target and the low-level starting point are known for a wide range of interesting problems. In those cases, the only challenge is to identify possible intermediate variables and pathways to connect the two models. However, in examples such as the Turing machine in the GoL, or interactions between species of living organisms in nature, or the study of the Boids model in this thesis, many of the properties that are observed begin with a hypothesis founded on a combination of creativity, and observations about the self-organized structure. For example, it is not a stretch of the imagination to suppose that flocks have length and slope because the self-organized equation for each pair of birds contains a relative displacement vector which is two dimensional. It is also not a stretch to suppose that flocks have a position, speed, and heading, because the components of the flock clearly do, and the self-organization does not impede the boids from continuing their forward motion. The first step to emergent behavior identification is grounded in observations of properties that are familiar, and then extrapolation to the unfamiliar when the self-organized object somehow deviates from expectation. Fortunately, in that case, too, it is possible to obtain information from the structure of the object, because whatever new behavior it exhibits ought to be grounded in some perturbation of the structure or change in the properties of the components.

5.1.3 Model Complexity Calculations

In number theory and computer science, the complexity of an *algorithm* is a well-established concept. However, the notion of the "difficulty" or "complexity" of a model (mathematical equation) is not so straightforward. "Evaluating Derivatives," a textbook by Griewank and Walther on algorithmic differentiation [226], reminds readers that mathematically compact expressions can be deceptive.[197] The authors then list multiple reasons why the complexity of the algorithm implemented to evaluate a model can vary: named intermediates, joint allocations, program branches, and iterative loops. Many of those features are often explicitly omitted from models (e.g. the final equation for computing the determinant of a matrix may mask the number of pivots performed, which would be implemented as program branches that severely affect algorithm performance).

[197] In the Prologue to the book the authors tackle a popular misconception about mathematical functions that engineers often think can be evaluated "explicitly," or "symbolically," rather than "numerically." Engineers might occasionally forget, for example, that transcendental functions are short-hand expressions representing infinite power series that can only ever be approximated (and doing so is often perfectly acceptable [333]).

Then, on the matter of derivatives, the authors immediately reject the use of the familiar secant equation despite the fact that every modern calculus book uses those equations to introduce the concept of the derivative. Not only is it less accurate, it is also slower than algorithmic differentiation. Finally, on the subject of the equations themselves, it seems equally "difficult" to write "$y = e^x + 2$" as it is to write "$y = x^2 + 2$" despite the fact that the transcendental terms represent irrational numbers (they are infinitely long).

This thesis requires a quantitative way to argue that one model is more "difficult" or more "complex" than the other.[198] Fortunately, such questions have already been asked. Rather than directly measuring the length of the equation (as one might do using Information Theory when comparing information content on the basis of Kolmogorov Complexity[163]), one can measure the difficulty of solving the equation by counting the number of atomic operations a Turing Machine would need to evaluate the mathematical expression. In research by Bowein and Bowein, the authors write,

> *(1) How much work (by various types of computational or approximation measures) is required to evaluate n digits of a given function or number? (2) How do analytic properties of a function relate to the efficacy with which it can be approximated? (3) To what extent are analytically simple numbers or functions also easy to compute?* [227]

It is customary to characterize the complexity of an algorithm based on the leading term of the equation that describes its asymptotic growth in the number of operations it performs with respect to the size of the input the algorithm receives. This complexity is expressed using "BigO" notation (the interested reader is referred to [228] [229] for more

[198] Additionally, this thesis needs a direct link between the self-organized object to the properties it acquires as a result of that self-organization, and then to the mechanism by which that property can be affected. This effort attempts to align all three at once.

information). However, since the output of a mathematical function is a number, and computer algorithms represent numbers in binary, it is more appropriate to use the bit complexity of the algorithm. Bit complexity also uses BigO notation but the "O" is modified to denote this subclass.[199]

There are two approaches one can take when computing time complexity using asymptotic BigO equations: (1) base all C_T calculations on evaluating equations up to a given number of digits (such as single or double precision numbers),[200] or (2) base C_T calculations on evaluating equations to arbitrary precision.[201] The first approach suffers from the drawback that one must prove that the algorithm being used to compute that number is the most efficient algorithm possible, which is difficult to prove. There are usually many possible algorithms for the same problem, all of which vary in their average performance and worst-case performance. Usually, an algorithm that is superior in its average performance is inferior in its worst-case performance. It is not typically possible to compare such algorithms using a single measure of efficiency.[202] This thesis will utilize the second approach because it is possible to derive theoretical estimates that abstract-out a variety of implementation-specific issues. Furthermore, such comparisons are well established in the computer science literature. Finally, the second approach is more appropriate for the time complexity of models because models represent exact results, not approximations. However, the second approach has the drawback that comparisons cannot be more specific than the use of BigO notation. That is, it becomes impossible to compare

[199] Since only bit complexity is used in this thesis, no distinction will be made in the notation.
[200] This is the approach taken by AI Feynman [295]. See also the Appendix.
[201] This is often referred to as "multiprecision arithmetic" in the literature.
[202] Given the level of abstraction at which this thesis is operating, it appears one can make the argument that it is better to simply replace every algorithm with a massive lookup table, and reduce the entire discussion of complexity to a comparison of lookup table sizes. This option will not be explored.

two algorithms using a fixed number (one cannot say "adding two 5-digit numbers requires 11 steps").

Derivations of the bit complexity for multiple operations are provided by Borwein and Borwein [230] [231], many of which are asymptotically equivalent to the derivations by Schönhage, Grotefeld, and Vetter [232].[203] The time complexity of the cube root is discussed in [233]. Schönhage et al show that addition and subtraction are O(n) where n is the number of bits in the binary number representing the full number (e.g. $C_T = 11n + 68$ [232]).

Table 2 – Time complexity of various operations and functions

Operation / Function	Time Complexity
+, -, $\|x\|$	≤ O(n) [232]
×, ÷	$O(n \log n) \leq M(n)$[204] $\leq O(n^2)$ [230] [232] [234]
$\sqrt{x}, x^2$	0.5×M(n) [232]
$x^3 = x(x^2)$	1.5×M(n) [230]
$x^6 = (x^3)^2$	2×M(n) [230]
e^x, $\ln(x)$, $\sin(x)$, $\cos(x)$, $\sqrt[3]{x}$	O(M(n) log n) [231]

[203] The current Wikipedia page [235] on the subject is well documented and deserves mention.
[204] M(n) is the time complexity of multiplication. Multiplication, and division are of the same time complexity (linear speed-up theorem notwithstanding).

Multiplication, implemented the naïve way, has complexity $O(n^2)$ (see Section 6.3 of [230]). However, it was just recently proved [234] via galactic algorithm[205] that multiplication of incredibly long numbers can be performed in O(n log n) operations (this was a long sought-after theoretical minimum). These results are summarized in Table 2. The claim that squaring a number is faster than multiplying two different numbers is based on the RSQU algorithm (8.1.33) in Chapter 8 of Schönhage's text [232]. Borwein and Borwein gives exponentiation as the 6th exercise in Section 6.2 of [230], with an emphasis on minimizing the number of multiplications needed. However, that exercise precedes their discussion of fast multiplication (which then cites Schönhage).

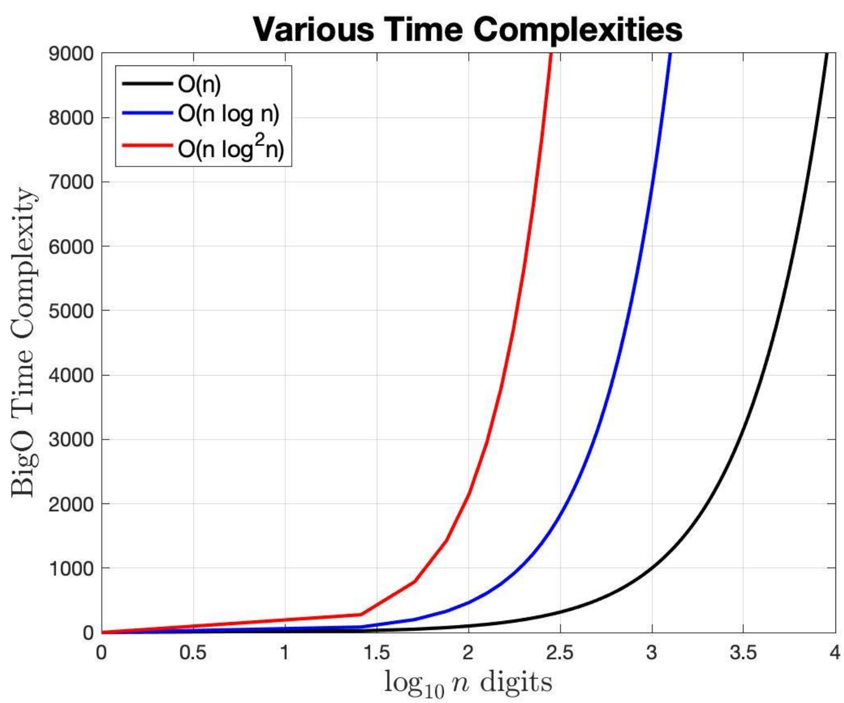


Figure 23 – Time complexity for n-digit number by operation / function

[205] This is a technical term for algorithms applied to problems that are so computationally expensive they are too impractical to implement (requiring more data than could be generated on Earth).

The coefficients given for exponentiation in Table 2 are based on a combination of using RSQU with fewer multiplications inspired by the procedure in Borwein and Borwein.[206] Unlike exponentiation, Table 2 treats all transcendental functions as having the same C_T. Several references on this subject point to Knuth [235] for more information, but this author does not have the background required to properly assimilate the body of information contained in his references in a timely manner.

The asymptotic behavior of three time complexities given in Table 2 are depicted in Figure 23 (the complexity of multiplication, M(n), is taken to be O(n log n)). Although the level of detail provided by Schönhage et al. is ideal for finite-precision arithmetic, it is hard to find that level of detail in the literature for every function utilized by SISSO. The constant in front of the leading term would be useful in distinguishing between similar functions but those leading coefficients are not widely available (and suffer from a kind of subjectivity that will be discussed later in this subsection). If the results generated over the course of this research require time complexity calculations that provide greater granularity, the approach and results will be discussed in the relevant subsection for that experiment. Finally, note that the time complexity of the models presented in this thesis will account for simplifications by substitution. For example, $f = (x+y) + e^{(x+y)}$ and $f = (y+x) + e^{(x+y)}$ are both treated as $f = z + e^z$, $z = x + y$, so that only two additions are performed rather than three. Here z is a temporary variable (i.e. a named intermediate, as discussed above). Temporary variables produced this way will not be counted against the space complexity of the model.

[206] This leads to the result that $C_T(x^4) = C_T(x \times y)$. Two squaring operations have the same complexity as one multiplication of two numbers, x and y.

Note that an analogy between the *time* complexity *of an algorithm* and the *time* complexity *of a model* is challenging to define for three additional reasons. First, model time complexity can take on two meanings when the system of equations contains integrals or derivatives. Take a partial differential equation, for example: Although the number of dependent variables remains the same, the time complexity can either be the time complexity of the original equation (which requires somehow quantifying the complexity of differentiation and integration), or it can be the time complexity of the analytical solution to the equation, which may be an infinite series. Furthermore, each representation may introduce new sources of numerical instability while eliminating physically meaningful sources of instability, as cautioned by Schmidt [156]. It truly appears to be an inescapable quality of complexity, that in any controlled study of emergent behavior, every attempt to define, measure, or otherwise control one source of complexity comes at the expense of observing or understanding another source of complexity.[207,208]

Secondly, most problems do not have a one-to-one correspondence to the algorithms used to solve them (e.g. the myriad sorting algorithms). Many algorithms can have their operations reorganized in order to become more efficient, except that rewriting algorithms often takes the form of trading space complexity for time complexity. There is an analogy for this in some engineering problems (such as decomposing large or challenging optimization problems into smaller, related problems), but model decomposition (exchanging highly nonlinear equations for more numerous, simpler

---

[207] Although Boyd was referring to human beings (which he characterized as open systems – a correct characterization within the context of SE), his comments bear resemblance to this argument: "Interaction permits vitality and growth while isolation leads to decay and disintegration" (emphasis in original) [186].

[208] There are contemporaries to this research that also factor time complexity into their model selections (unfortunately, time does not permit a comparative study): AI Feynman 2.0 uses Bit Complexity [346], while SINDy counts the number of terms in the equation [248], and both use a Pareto Optimal approach.

equations) is very risky when it comes to studying emergent behavior because it introduces new sources of aleatory and epistemic uncertainty. Furthermore, algorithms for arithmetic operations or the evaluation of transcendental functions are often based on the number of atomic operations performed. Such calculations are only possible because the numbers being added have a fixed number of significant digits (i.e. the number can be stored as 16, 32, or 64 bits [159]). Real numbers can be irrational[209] or arbitrarily long, making the number of operations unknowable *a priori*. Thus, to be analogous to algorithmic time complexity, this thesis will use the asymptotic multiple-precision time complexity of algebraic operations and elementary functions as the basis for its comparisons. This assumption will limit the predictions made by this thesis in one important way. Many algorithms for performing an operation such as multiplying two floating-point numbers, have the same worst-case performance, but vary on average, or vary by a constant, which is masked by big-O notation. This can cause two different models to appear to have the same computational complexity.

Thirdly, there's the linear speed-up theorem [236]. This ties back to Kolmogorov Complexity from information theory. Just as the Kolmogorov Complexity cannot be uniquely computed for any description (only the mean is estimated) there is no unique representation for a model, and thus no unique ordinal system of comparison for closely related operations. In other words: it is easy to compare multiplication to addition, and transcendental functions to multiplication, but there exists a Turing machine for which e^x is faster than sin(x), and an identically constructed Turing machine for which sin(x) is faster

[209] Not to mention other classes of numbers such as complex numbers or quaternions, which are categorically different yet nevertheless well-defined objects.

than e^x by a constant multiple. If we select a unique Turing machine (e.g. the one where e^x is faster than sin(x)) and perform all of our calculations on that machine, we would obtain one set of C_T values that can be sorted so that our various models are ranked consistently. In general, however, that ranking is not perfectly universal (i.e. the linear speedup theorem cannot make multiplication faster than addition, but it can make division faster and multiplication and vice versa). In terms of the Pareto Front plots that will be demonstrated in CHAPTER 6, where the horizontal axis is C_T, this means that any point on the plot can move left or right within its complexity "bracket" (the range corresponding to its asymptotic growth).[210] That, in turn, means that any pair of dominated and dominating points can translate within a complexity bracket, potentially causing them to swap roles (the dominated point becomes dominating). Thus, the linear speed up theorem introduces a form of unknowability analogous to the limitations of Kolmogorov Complexity. This is at once reassuring and frustrating. It shows that this approach of computing C_T does not violate a basic principle of Information Theory, which is reassuring. However, it is frustrating because it results in a C_T estimate that is somewhat vague. This ambiguity is partially remedied, however, by examining the entire Pareto Front with a large sample of points (just as one would want a good estimate of the Kolmogorov Complexity). There are three cases where the linear speedup theorem will have no effect: (1) If the difference in performance between models is sufficiently great, such as one model having transcendental terms while the other is purely algebraic, or (2) if the type of highest-order operation in

[210] Though relevant, determining precisely how much each model would slide by some linear speedup is outside the scope of this thesis.

multiple models is the same, such as two models whose transcendental terms are e^x, or (3) cases where the form of the functions are the same.

A second notion of difficulty comes from the number of variables needed to complete the equation. Clearly a model with more variables is more difficult to evaluate (all else equal).[211] As stated in Section 2.2, the number of variables in an equation will be referred to as the ***space complexity*** of the model. While this might seem obvious, the mapping of space complexity between a model and the algorithm used to solve it is not 1:1.[212] Moreover, computer scientists generally measure the space complexity of an algorithm in terms of the extra variables "over and above the space that is needed to store the given input" [159].[213] This distinction is critical for two reasons: (1) it is often possible to reduce the time complexity of an algorithm by increasing its space complexity (e.g. computing x^n [232]),[214] (2) the physical architecture of a machine, as in its cache and physical memory, directly impact the efficiency of the algorithms implemented on it.

To properly associate the *space* complexity *of the model* to the *time* complexity *of the model*, this thesis takes the space complexity of a model to be precisely the number of *dependent* variables needed to write the system of equations that capture the desired behavior.[215] Computer scientists scale algorithm complexity as a function of the number of inputs to the algorithm. The analogy in modeling, is that model complexity scales with the

---

[211] Some authors use the phrase "parsimony" or "parsimonious model" to refer to finding the model with the fewest variables.
[212] Recall from the discussion in Section 2.2 that there are infinitely many ways to write a model.
[213] Engineers typically refer to this as "temporary variables" in our codes.
[214] This is an example of the aforementioned named intermediates.
[215] The underlying assumption being that given the infinite set of possible descriptions for a phenomenon, there exists a boundary containing the model with lowest possible space complexity, the model with lowest possible time complexity, and all Pareto Optimal models in between. That is, while our models can be infinitely complex, the extent to which they can be simplified is bounded.

number of system components being modeled since each component contributes one or more equations to the system of equations that needs to be solved (independent variables such as space and time do not scale with the number of components).[216] Therefore, when two or more components exhibit a pattern of behavior (as discussed in Section 3.1) the equations that those components contribute to the overall model become at least partially redundant. Parts of the system of equations can be replaced with a smaller set of equations that captures the essential features of the pattern (as illustrated in Section 3.3). The amount of data compression (the space complexity change), then, is the difference between the number of component-level property equations that were removed from the system, and the number of pattern-level property equations that were added to the system (in principle[217]). Note that this applies even to differential equations.

Another issue with space complexity calculations is upward causation. If all information of lower level behavior is retained, then all high-level models actually have greater space complexity than their lower-level counterparts. For example, the position of a flock is typically computed as average positions of boids. Therefore, in any method where the simulation is run to track the boids and additional mathematics is performed to track the "flock-level" behavior, then the total number of variables and equations has increased. To treat the high-level property as an idealized property (Section 2.2.2), information about the individual components must be discarded. This almost always guarantees some amount of information loss (e.g. multiple configurations of boids can produce the same center of gravity, and thus the same flock position). Nevertheless, it is standard practice across all

[216] In a sense, this model complexity scaling is more closely related to program complexity, since the total memory usage of the program scales with the number of components it is simulating.
[217] Recall that, in general, this is lossy compression. Quantifying that loss remains to be rigorously explored.

modern disciplines to discard this information at higher levels of abstraction, and yet they serve their purposes well. For example, the projectile motion equation describing the flight of a free-falling object does not take into account the exact position of every atom inside the object, nor does it need to so long as the underlying assumptions are satisfied. The moment the position of any one atom in the object becomes important, the projectile motion equation loses most of its utility. In this way, we see that nearly all compression is lossy even when the pattern is perfect. So when the space and time complexities of system-level equation are compared to component-level equations, it must be remembered that information is lost in that process, and such a loss is justified whenever that information is negligible with respect to predicting the desired behavior of the higher level object.[218,219]

5.2 Self-Organization Detection Tool Workflows

The method in this thesis is geared towards simulation data, and so its principal input is the time series data generated by that simulation (Figure 24). As discussed in Sections 3.1-3.3, identifying self-organization in a simulation can be achieved by performing pattern recognition on relative values (such as relative distance). Patterns are picked out from that data by using the Fast Fourier Transform (5.2.1), and Fourier Series (5.2.2), as well as a user-defined time interval (a.k.a. window) and data regarding component interactions. When one or more relative properties of two or more interacting components exhibit concurrent periodic signals over the same time window, that group of components can be classified as a self-organized system. In this thesis, all systems will be classified based on

[218] This concludes the "something from nothing" discussion started in Section 3.2, continued through 4.3.2.
[219] This raises the obvious question: how does one guarantee that low-level equations and high-level equations are written such that an emergent equation bridging that gap can be written? Furthermore, if information is lost between levels of description, it is very possible that the crucial information needed to bridge models between levels was lost in the process! This is future work.

their geometry because only kinematic equations are considered (as opposed to dynamic and thermodynamic equations). The set of variables that are needed to classify a set of components as a self-organized system (the ones resulting concurrent periodic signals) is always problem-specific, and so this process requires at least some human judgment.

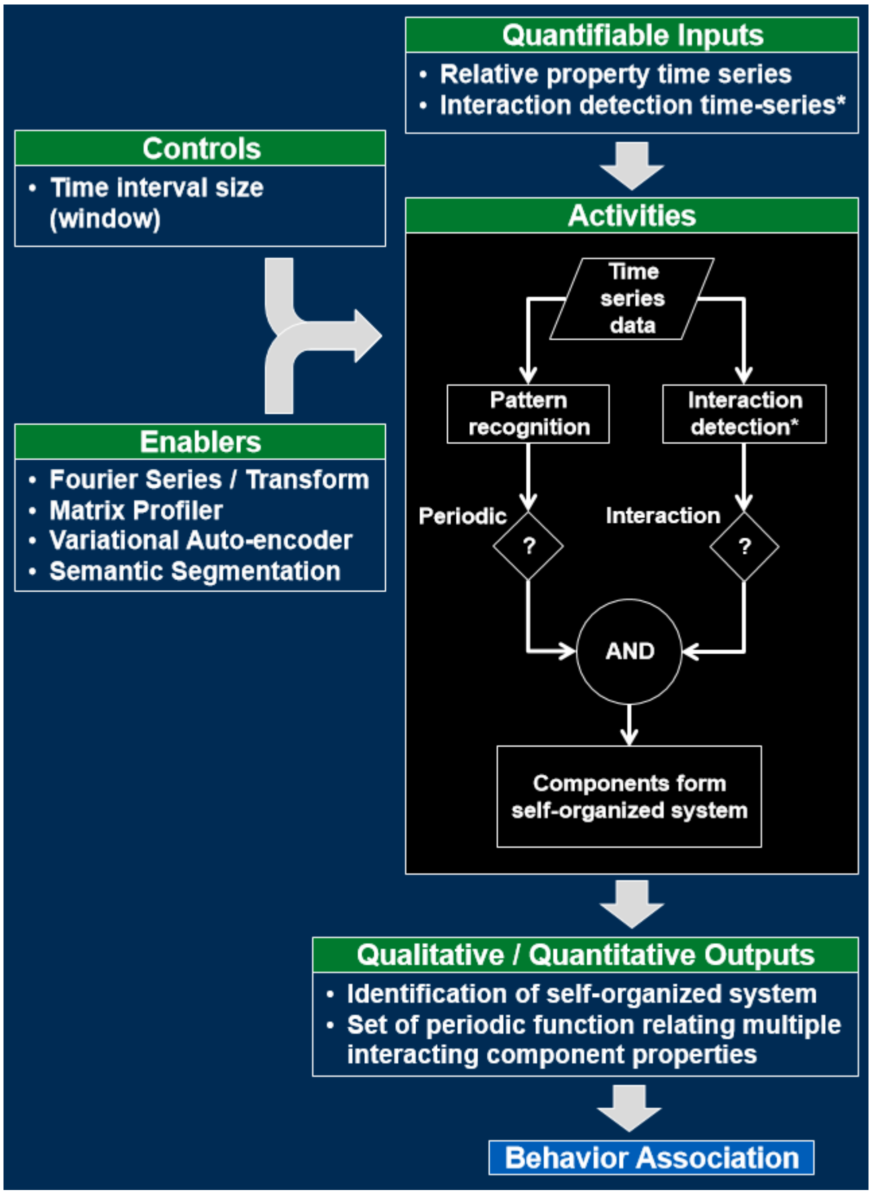


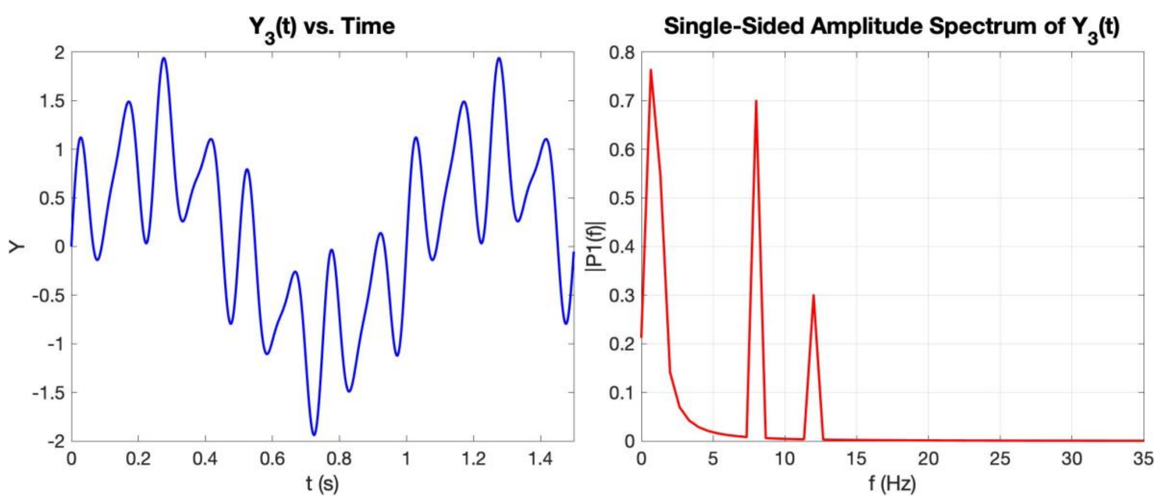
Figure 24 - Input-Process-Output diagram for pattern recognition step[220]

[220] The interaction detection process (marked with *) was recognized as necessary during the experimentation process, as illustrated by the document narrative. It is included her for completeness.

A very sophisticated tool that can dramatically simplify the self-organization identification process is the Matrix Profile [237] [238]. Although that tool is not needed for this thesis, it will likely be indispensable for extending the method developed in this thesis to very large data sets due to its ability to pick out similar periodic signals over large time intervals. Variational auto-encoders and semantic segmentation are also important tools for pattern recognition, due to their ability to associate patterns under a variety of coordinate system transformations with human-interpretable classifications (the interested reader can begin their literature review with [239] [240]).[221] These enablers are listed in Figure 24.

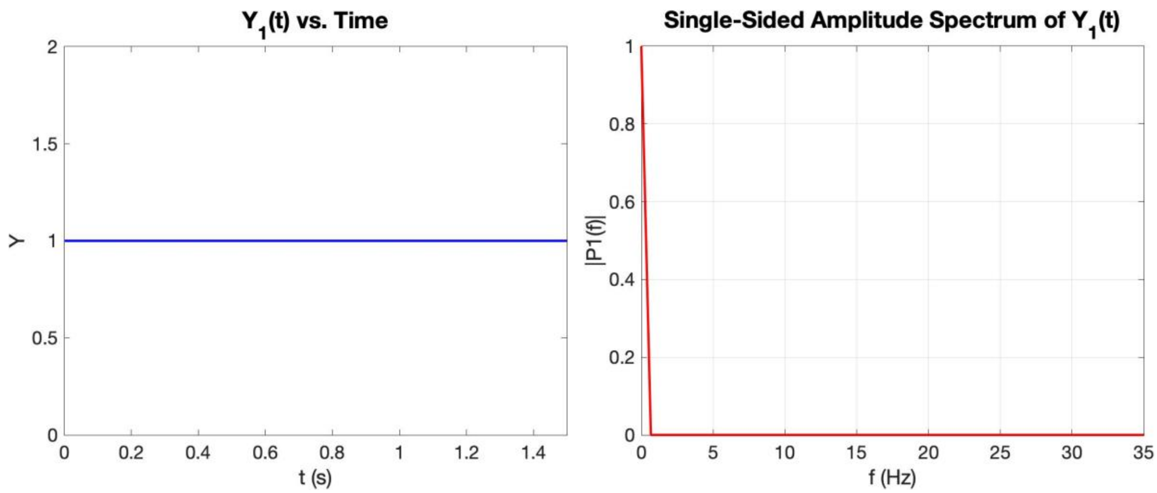
5.2.1 *Tool 1: Fast Fourier Transform (FFT)*

The FFT converts time series data into frequency-domain data, which significantly simplifies the identification of a periodic behavior. A simple implementation is provided on the Matlab website [241], and is used here.



(a - TS) **(a – FFT)**

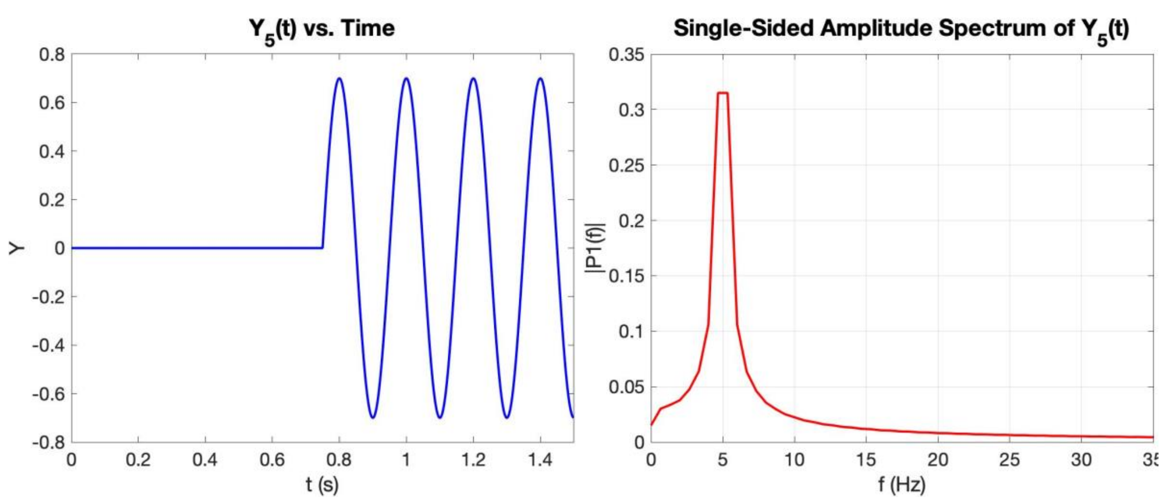
[221] These advanced tools are one way to deal with the combinatorial explosion of relative property time series.



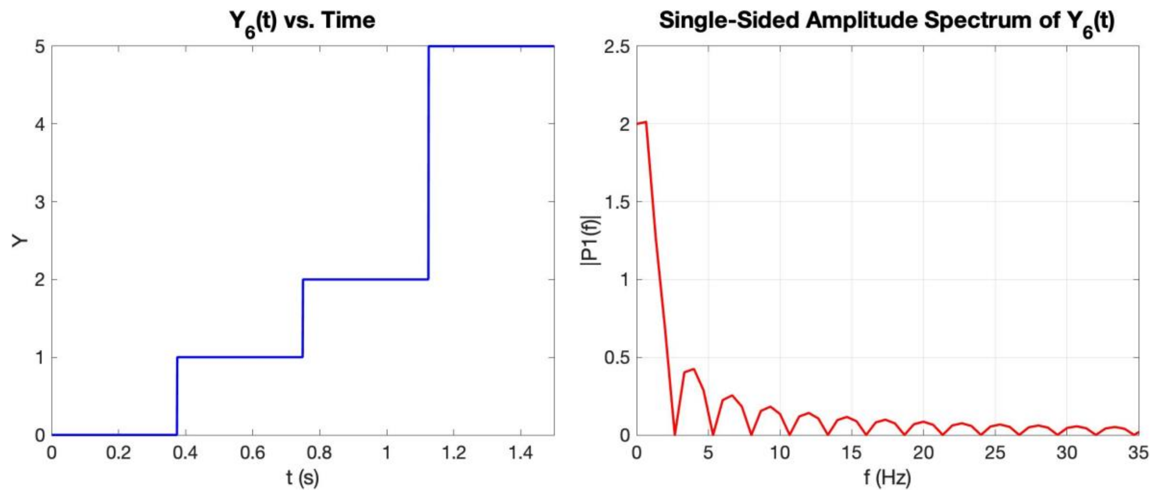
(b - TS) **(b – FFT)**

Figure 25 – Fast Fourier Transform (-FFT) applied to "clean" time series (-TS) data (a) sum of sine waves, (b) constant non-zero signal

Figure 25 illustrates two examples of periodic functions and plots of their corresponding FFT. In the first case, (a-TS), three sine waves are added together. Their respective frequencies correspond to the three peaks shown in (a-FFT). In the second case, (b-TS), a single non-zero constant function is shown, which produces a single peak at 0 Hz (b-FFT). Most smooth, non-periodic functions will produce similar peaks.



(a - TS) **(a – FFT)**



(b - TS) **(b – FFT)**

Figure 26 – Fast Fourier Transform (-FFT) applied to "mixed" time series (-TS) data (a) piecewise defined signal with sine wave, (b) step function

The difference is that the constant function will only produce a single non-zero value above 0 Hz, whereas the other functions will produce a single-tail distribution of frequencies that peak at 0 Hz.

Figure 26 shows examples of periodic signals that are not uniform in the given time domain. In the first case, (a-TS), the first half of the time interval is an all-zero signal,[222] which does not produce a peak in FFT plots, followed abruptly by a sine wave. The frequency of that sine wave is indicated by a single peak in the FFT plot (a-FFT). In the second case, (b-TS), a step function is given. Since each of the three steps have a frequency of zero, a single peak appears above zero in the FFT plot (b-FFT). These examples reveal a few limitations of relying solely on FFT. First, it is impossible to distinguish spatially or temporally distributed signals from one another if they have the same frequency. Those signals will simply aggregate into one large spike above their mutual frequency in the FFT

[222] In most applications this is typically interpreted as the absence of a signal. That is not the case here, as will be discussed later in the section.

plot (as shown in b-FFT of Figure 26). Another issue with FFT is depicted by the various little bumps in (b-FFT) of Figure 26: these bumps are numerical artifacts of the discontinuities in the step function.[223] In the case of data containing vast amounts of interactions, those bumps would be comingled with other time series, which are known as "noise" in the signal-processing literature. If a time series is too noisy, or, as in this thesis, if the number of signals being simultaneously tracked becomes very large, it will be impossible to distinguish meaningful spikes in frequency from background transient data.[224] Nevertheless, for small data sets with no other source of noise and an adequate windowing procedure (see below), a FFT is a quick and easy way to isolate self-organization from time series data of relative properties. Moreover, one must carefully select the relative metric to track in order for a signal to appear in the FFT plot. For example, one can detect that two boids are flying parallel to one another by tracking the relative heading ($H_i - H_j$) or the normalized dot product of their vector velocities ($\vec{V_i} \cdot \vec{V_j} / \|\vec{V_i}\| \|\vec{V_j}\|$). The difference here is that when the boids fly in a linear flock $H_i - H_j = 0$, which will *not* appear in the FFT plot, while $\vec{V_i} \cdot \vec{V_j} / \|\vec{V_i}\| \|\vec{V_j}\| = 1$, which *will* produce a visible peak in the FFT plot. Finally, the number of measurements taken during a particular time interval (sample size) influence the quality of the observed peaks (distortions, and the appearance of small peaks due to approximation errors). ***Windowing*** is simply the process

[223] "Numerical artifacts" with regards to the goals of this thesis. In general, any non-constant, non-periodic curve will have non-zero spectra across multiple frequencies (i.e. the "artifacts" belong there). This will make it very challenging to observe flat regions of a space unless the right window size is chosen. Additionally, a scatter plot is easier to interpret than a line plot because the only non-zero points will be directly above the frequency of the signal (assuming a clean signal / good window), which leaves less room for error.
[224] "Noise" and background "irrelevant" perturbations, are nearly interchangeable concepts.

of breaking any data series into smaller intervals that cover the phenomena of interest (it filters out unnecessary, irrelevant, or otherwise confounding data) [242].

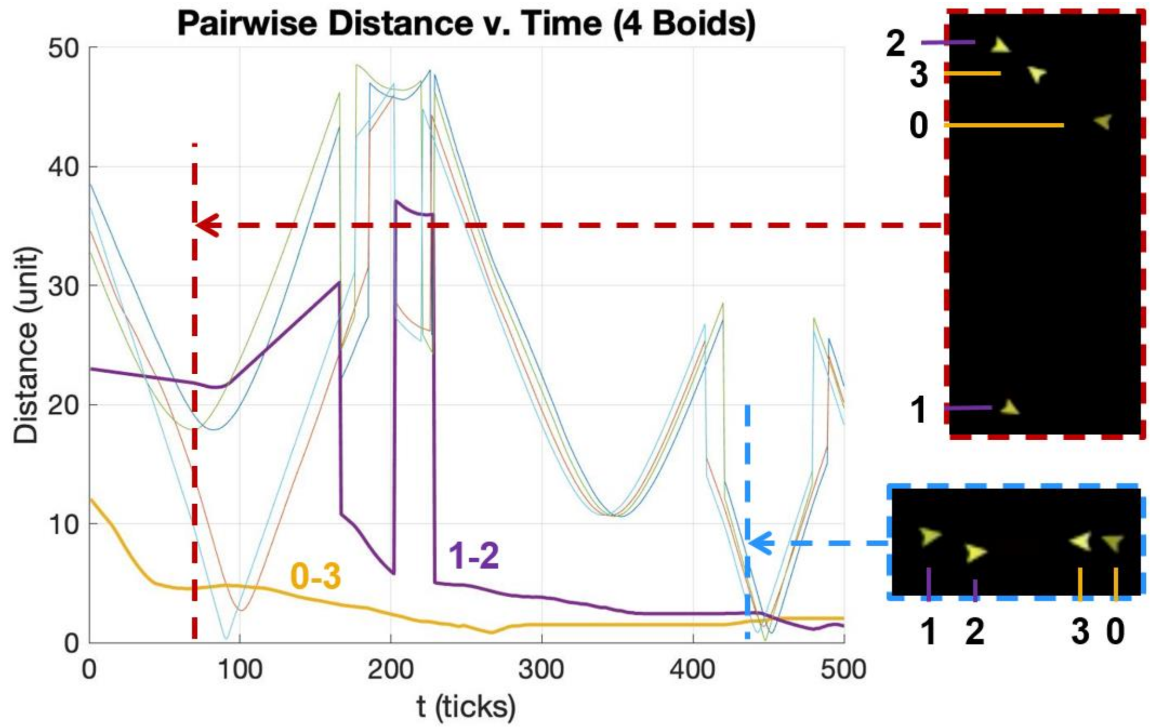


Figure 27 –Time-series plot of Boid relative distances and simulation screenshots at given moment in time

The width of the window is typically problem dependent and often requires some amount of experience working with the data. In the case of the Boids model it may be possible to derive a sophisticated window size rule based on knowledge of the boid properties (vision distance, updraft, speedup, etc.) but for the purposes of this thesis it is far more efficient to obtain a ballpark estimate by running a few simulations. Figure 27 shows the time series data for the distance between pairs of boids in a simulation containing 4 boids, as well as snapshots of the distribution of boids in the simulation. There are six pairwise distance curves in total (4 choose 2). The two curves that are highlighted (yellow and purple bold lines) correspond to the pairs of boids that formed lines later in the simulation (e.g. boid 0 and 3 formed a 2-boid line).

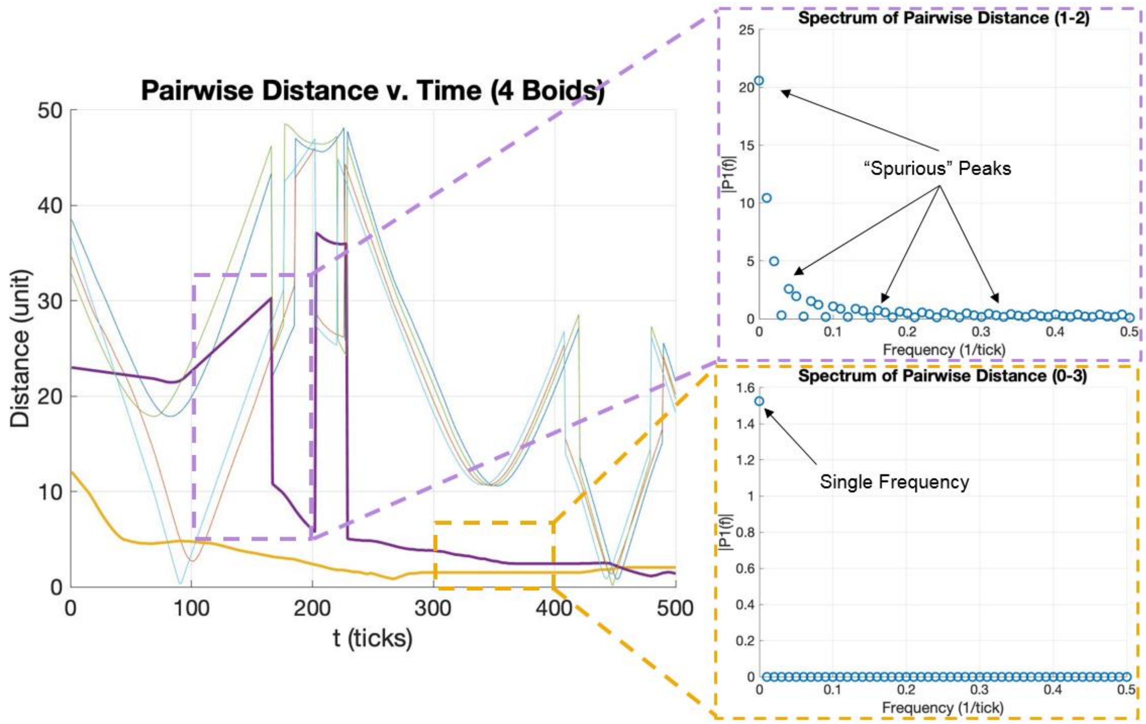


Figure 28 - Time-series plot of Boid relative distances with windows and FFT

Figure 28 shows the FFT plots that result from applying the FFT over two separate windows in the simulation (one for the purple time series, and one for the yellow time series). Observant readers will note that a correction has been applied to the distance measurements of self-organized flocks to account for sudden jumps caused by the periodic domain.[225] As is clear from those figures, when the boids organize into a single flock, the corresponding FFT contains one large peak. This provides a numerical approach for

[225] As boids fly across one boundary, they disappear from one side of the domain and reappear on the other side. This causes naïve distance measurements between boids to display step-function behavior. Fortunately, they are piecewise continuous, and so corrections are straightforward.

distinguishing self-organization from disorganized behavior among agents, and can be performed for any set of properties.

5.2.2 Tool 2: Fourier Series (FS) Curve Fitting

A second approach to identifying a region of periodic behavior is by fitting a FS to the data. In order to reduce the occurrence of false positives (as mentioned in Section 3.1), this will be performed in conjunction with the FFT and a windowing procedure. FS are a natural fit because they can fit most periodic functions of interest to engineers [243].[226] Figure 29 depicts a notional example of a pairwise relative property (e.g. relative heading or distance) that at first follows some arbitrary nonlinear trend and then converges to a periodic time series. A window of 8 ticks is placed on the time series shortly after it appears to stabilize. The equation for the pairwise property depicted inside the window of Figure 29 corresponds to the following equation:

$$P_i - P_j = 2 + \sin(3t) + \sin(5t) \tag{3}$$

The absence of other dependent variables on the right hand side of Eq. (3) is the evidence that the property, P, of the i^{th} boid and j^{th} boid are locked into a pattern (here, a temporal pattern), which then indicates boids i and j have self-organized. If P were Euclidean distance, then the boids have a spatial arrangement that persists in time. In general, P could be any variable in any metric space.

[226] Attempts to generalize this will have to account for convergence issues (see Section 5.4 of [347]).

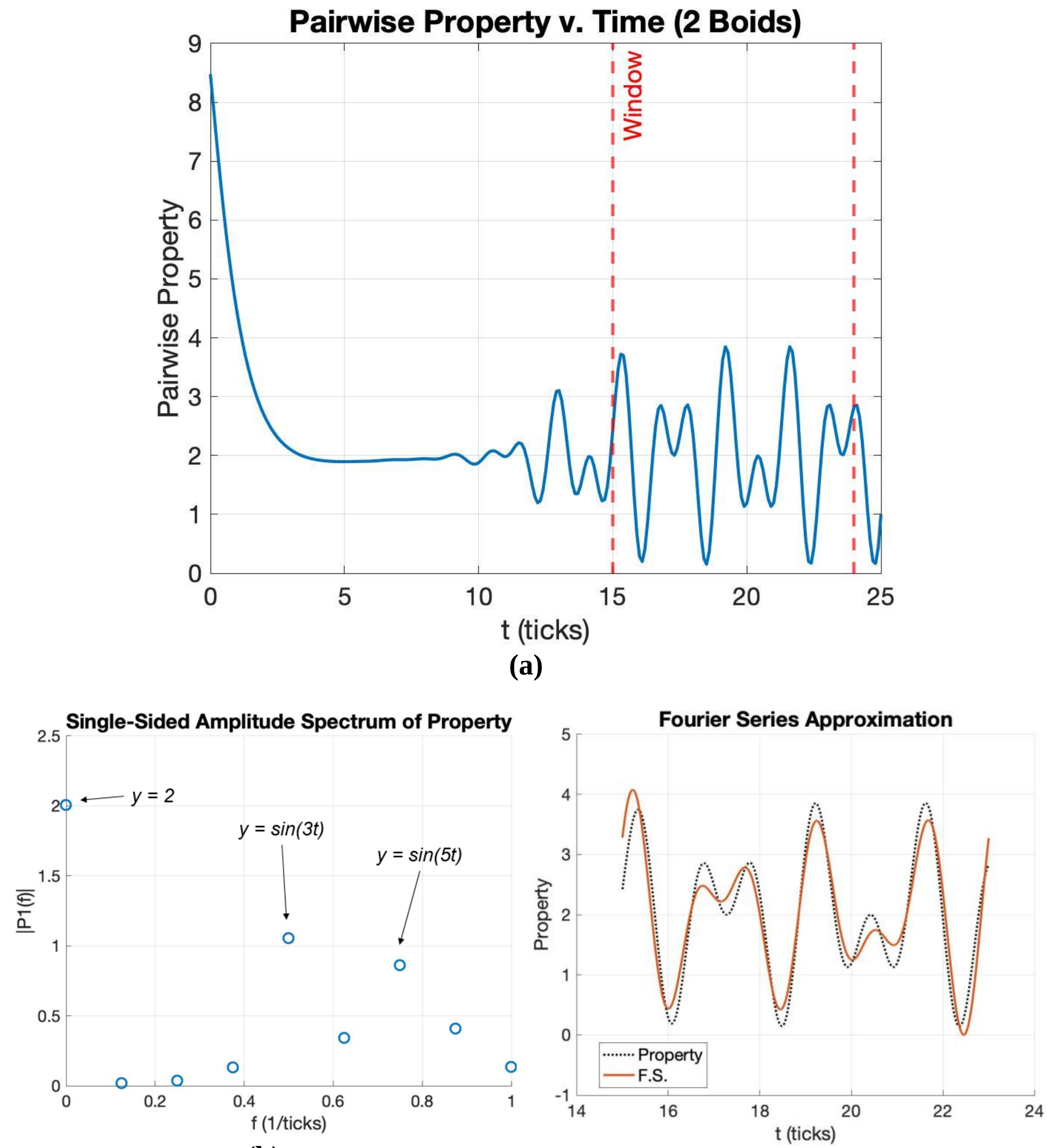


Figure 29 – Plots of pairwise property (a) time series with window on periodic behavior (b) FFT of window (c) 6-term FS of window

An FFT of the time series, shown in Figure 29(b), indicates three peaks that correspond to the horizontal line on which the periodic time series is centered (y = 2),[227]

[227] It is a coincidence that the 0-frequency peak happens to have a height of approximately 2.

and the two sine waves used to generate this portion of the time series. The peaks correspond to frequencies of 0.5/ticks and 0.75/tick. The true frequencies are $3/(2\pi) \approx 0.477$ and $5/(2\pi) \approx 0.796$. A 6-term FS is also shown in (c). Already it is clear that the FS is approaching the behavior of the property. More terms and/or a window sized to an integer multiple of the period[228] would be needed to improve the fit shown in this figure.

Note that the frequency peaks illustrated in the FFT plots can be used to accelerate the creation of the FS model by providing approximate initial estimates of the frequencies of each term as well as the number of terms. To be efficient, such an approach would require a nonlinear optimization scheme to tune the frequency and amplitudes of the periodic functions individually. The FS code used here works best for constant functions, and waves with integer-valued frequencies. A general approach will be considered for future work. The presence of the pattern will be verified by direct inspection of the movies recorded during simulation.[229] Figure 30 shows the overall self-organization detection workflow, and assumes that only one property is needed for self-organization detection. In practice this may not be the case. Linear flock detection requires three properties: a periodic relative distance, a periodic velocity dot product, and verification that each boid sees the boid in front of it. With some ingenuity, the third condition can be described as a periodic function.[230]

[228] That is, the value of the property at the beginning of the window equals the value of the property at the end of the window.

[229] It is for this reason the FS can be fitted using only RMSE. Without human judgment (and sometimes even with human judgment) autocorrelation and time series stationarity would have to be accounted for.

[230] The boid label can be sorted and concatenated into unique numbers to produce piece-wise constant representations of the vision field (e.g. a boid that sees three boids with labels [3,10,2] can use the number 100302 to indicate that it sees the group). This is an application-specific remedy.

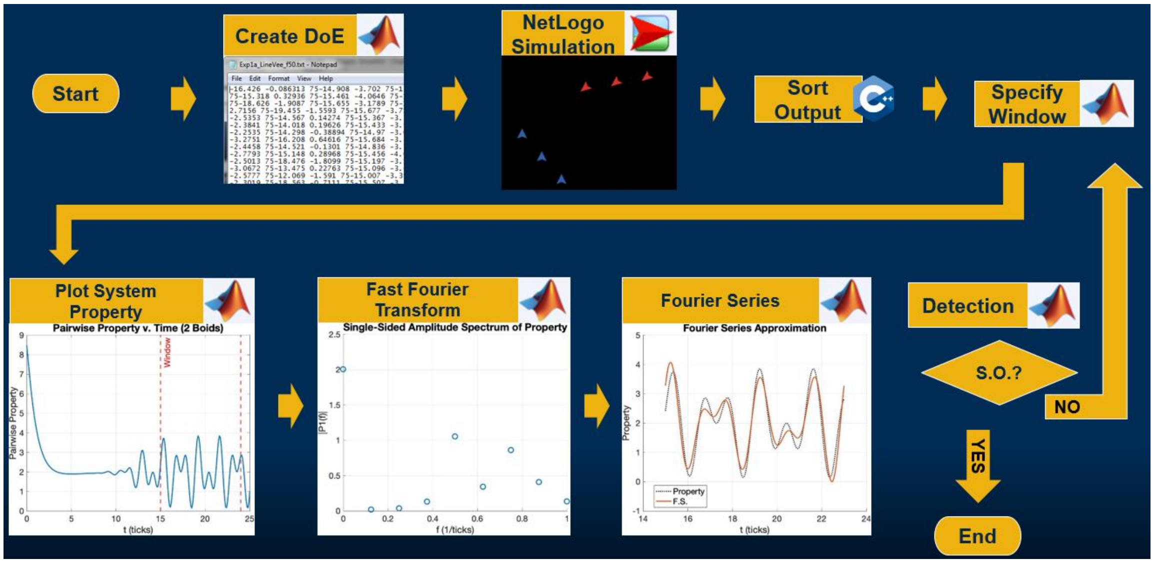


Figure 30 - Flowchart for self-organization (S.O.) data creation and identification

For simplicity, however, confirmation of mutual visibility will be performed with a script that directly checks and compares the indices of each boid. A linear flock is identified whenever all three conditions are met simultaneously. In general, any periodic signal would apply, and the data compression calculation would be modified accordingly (so long as the only new variable in the periodic expression is time, or some other independent variable, the compression calculation can proceed as indicated in Section 3.3). Generalizing this compression calculation is a topic for future study.

From Figure 30, note that Netlogo does not support space-fitting DoEs (full-factorial only) [244]. Therefore, a space-filling DoE is generated using Matlab and provided to the program via its file reading functions. The disadvantage to this approach is that Netlogo cannot simulate such DoEs in parallel. Also, Netlogo uses random order of execution so that no one agent receives preferential treatment [244]. Therefore, Netlogo output of agent properties needs to be sorted. This randomization will be more important when simulating adversarial flocks because it will cause adversaries to over/under-estimate their opponents speed and heading. It also means that simulations can never be repeated (unless the random order execution is manually overridden).

5.3 Behavior Association (Curve-Fitting) Tool Workflows

Before proceeding with the process, it is important to clarify a reason this thesis takes the approach that it does. Since the studies in this thesis are purely numerical, then in order to claim an emergent behavior has occurred without immediately making a circular argument, it is essential that two self-organized objects within the simulation interact, and that the properties they interact with be mined from the data (i.e. derived from the self-organized structure, etc., as described in CHAPTER 4), not coded into the simulation. Such behaviors can only be attributed to the self-organized group since the property being affected is computed using multiple component properties.

For idealized components with simple rules, the change of a component-level property only occurs during interactions. For boids, this means they only change heading and speed when see each other and/or enter the updraft region of another boid. Once the boids can no longer see each other (or if the conditions permit stable flight) the boid will continue in a straight line at constant speed until the next interaction. One significant difference between a boid-boid interaction, and flock interactions is that flocks will destabilize to varying degrees, and must re-stabilize before the pattern-recognition tools used in this thesis can numerically confirm their persistence.[231] That is, all interactions at

[231] Pattern recognition tools for open systems with time-varying composition, or systems that can undergo significant deformation without changing their fundamental nature are outside the scope of this thesis.

the system-level will cause perturbations in the structure of the system. This causes changes to properties of one or more components over time, which will then re-organize in response to the perturbation. Once the components re-stabilize, the self-organized structure can be detected using the approach described in Section 5.2. The time it takes for each system to stabilize can vary, as seen in Figure 31 (using a linear flock of boids as an example).

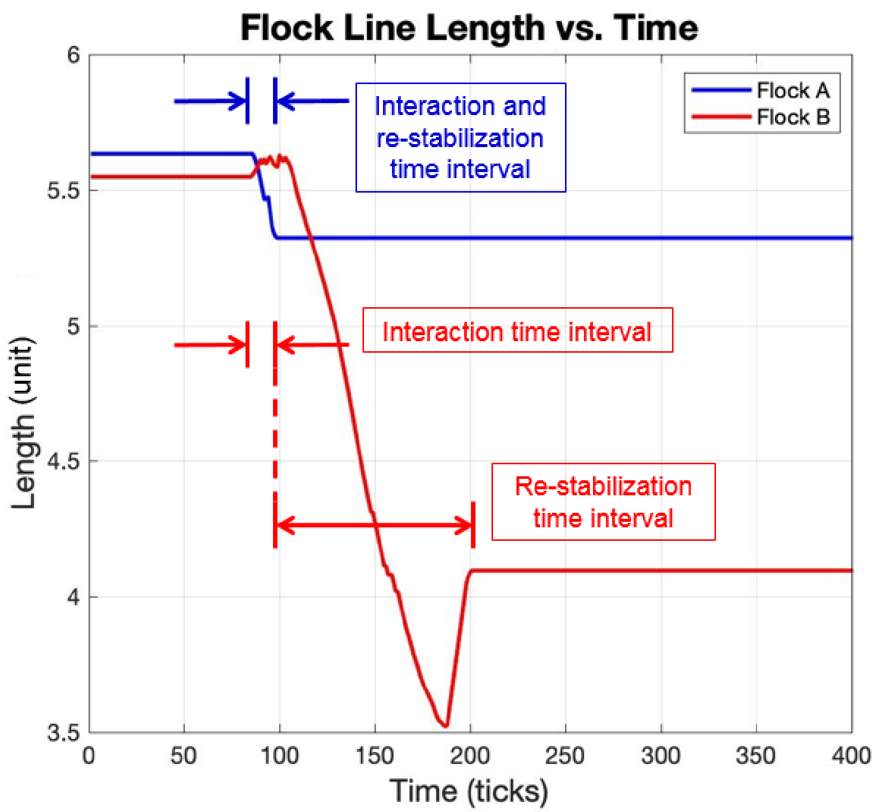


Figure 31 – Flock length change due to interaction and re-stabilization

In Figure 38, the time it takes for Flock A to stabilize is much shorter than Flock B. Furthermore, since interactions are one-way in the Boids model, it is possible for each flock to have different interaction time intervals (one flock may “see” the other for a longer period of time). To test against spurious regressions of one form or another, the experiments in Section 5.5 will cover five different time intervals: (1) Stable / Independent, the two flocks have not yet encountered each other, and all flock-level behavior can be calculated directly from boid-level behaviors, (2) Interaction: the birds in at least one flock see the birds of the other flock, and begin to maneuver accordingly, (3) Re-stabilization: the birds in the perturbed flock no longer see the birds of the opposing flock, and re-organize into a flock shape, (4) a combined interaction and re-stabilization time interval, (5) Full Time Interval: this includes an additional period of time after interaction to allow study of time intervals with an additional obfuscation of the data.

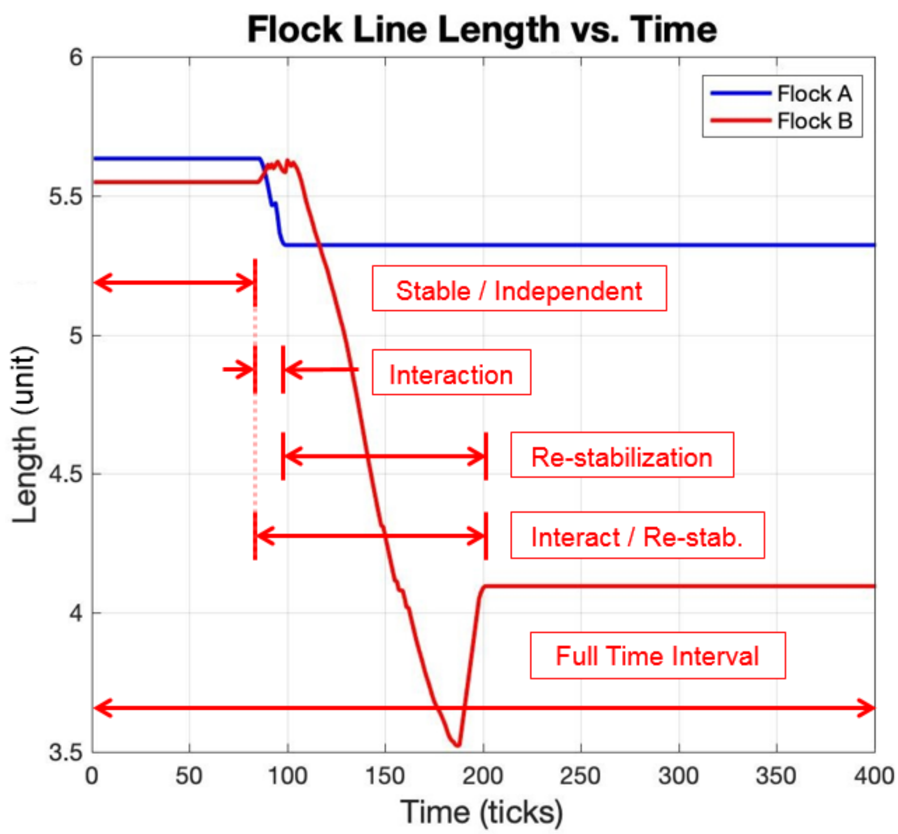


Figure 32 – Time intervals of interest for flock-level interactions

Since the qualitative and quantitative criteria for the existence of the system are contingent on pattern recognition, there are two ways to analyze the resulting time series data: (1) consider the system's properties before the interaction and after the interaction, while disregarding the intermediate transients. In this perspective, the only system properties that are ontologically justified are the ones that can be unambiguously associated with the stable system, since stable systems are the higher-level analogy of an idealized component.[232] So long as the system is not broken during its interaction, the transient perturbations caused by the interaction merely represent the minimum length-scale and time-scale over which the system operates, and is the minimum grid-size and time-step over which a system level simulation can be executed. This is common in engineering: if the beam in a truss buckles, the truss has failed and its load-bearing capability is altered or destroyed. Thus, the word "truss" loses some of its meaning. The name, properties, and meaning of a system, as well as the failure of the system to serve its purpose is intimately associated with the stability of its structure. Whether or not it exists depends on its ability to return to a recognizable shape after some perturbation. (2) Consider the system's properties before, after, and during the interaction. In this perspective, a model that predicts the time-evolution of the system's property from the onset of the interaction up to its re-stabilization is sought. This is also common in engineering, such as in irreversible processes, fluid dynamics equations, and solid-fluid interaction models. This second perspective is particularly important for open systems, and systems where some degree of structural evolution is within the scope of system definition (such as phase changes in a material).[231]

The behavior association process is depicted in Figure 33 assuming the implementation of the numerical criteria given in CHAPTER 4. In its current form, it is up to the analyst to manually derive a set of candidate properties for the self-organized system

[232] See Section 2.2.2 for the discussion on idealized components.

based on his/her own subject matter expertise.[233] With that information, the systems are simulated such that they interact.

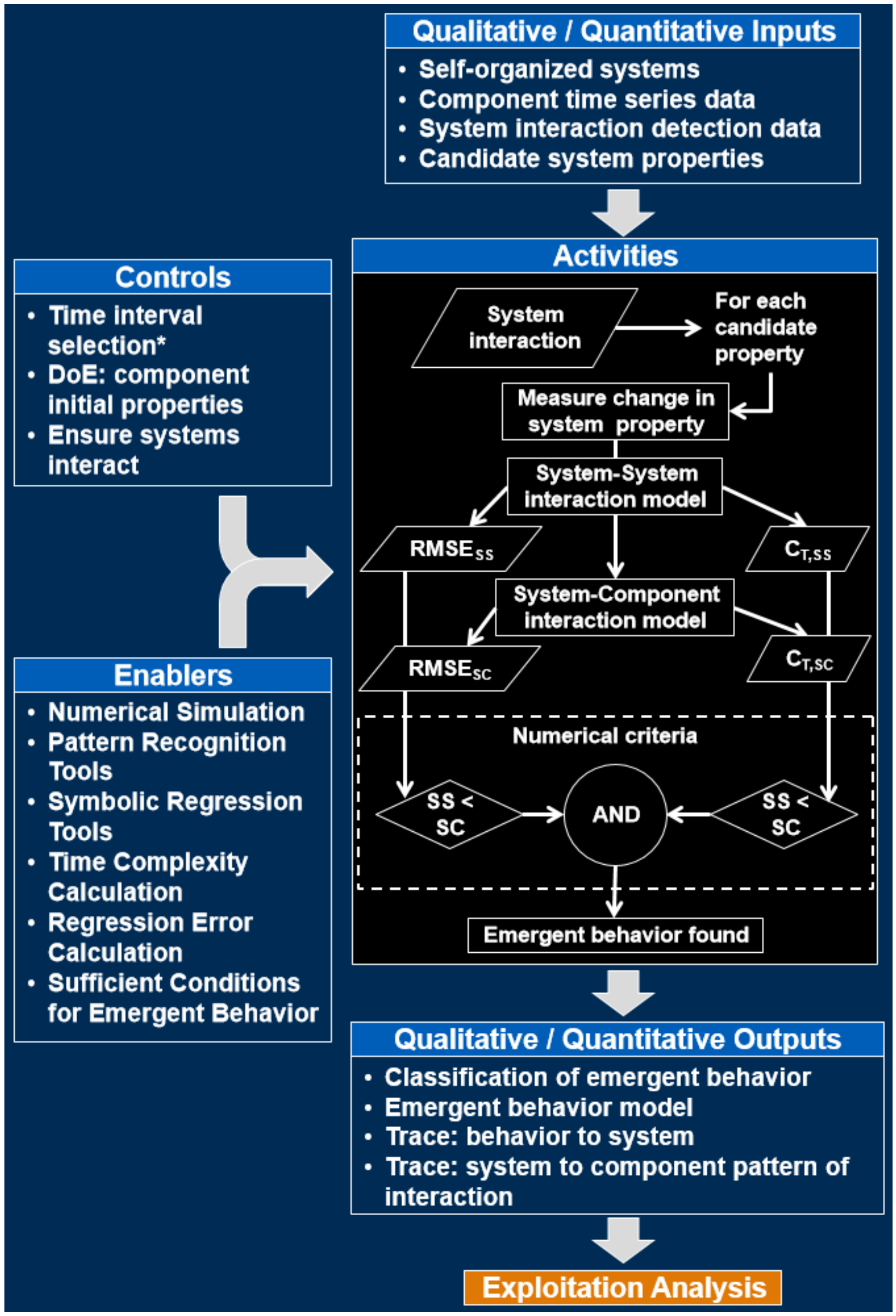


Figure 33 – Input-Process-Output diagram for behavior association step[220]

[233] Testing the numerical criteria for sufficiency requires implementing them as a "post-mortem" approach. Future studies can compare this to Szabo & Teo's work (discussed in Section 1.7).

Once a system interaction is detected (i.e. an interaction between components of two different systems), it will be possible to determine if the candidate property changes in a statistically meaningful way across multiple simulations. If the property does change, then the numerical criteria can be applied to determine whether the behavior is an emergent behavior (see Section 5.5 for more detailed information on the application of the numerical criteria, and the test for Hypothesis 2).

5.3.1 Time intervals using SISSO

This study focuses on how behaviors that occur over meaningful time intervals affect properties, as depicted in Figure 32. The duration of the interval, and the evolution of the property during the interval itself, are omitted from the data. Only the initial and final property values are studied.[234]

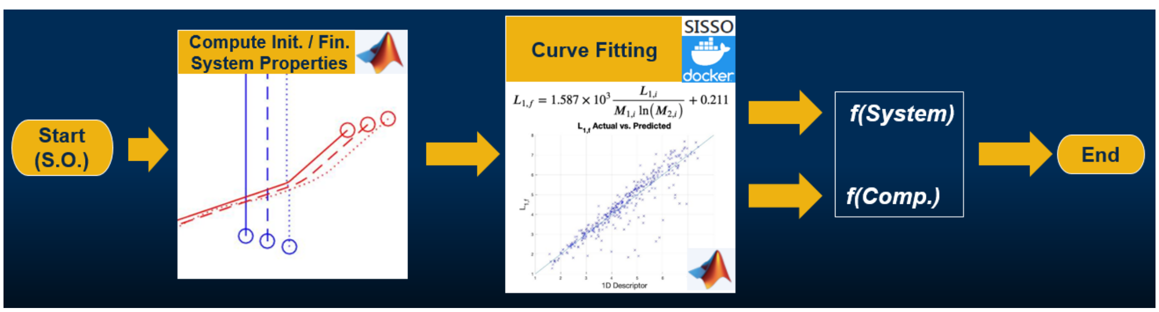


Figure 34 – Flowchart for flock interaction data creation and curve fitting

As shown in Figure 34, once self-organization (S.O.) is detected,[235] the simulation is continued until the self-organized object interacts with another object such that its structure is perturbed.[236] In Figure 34, two 3-boid linear flocks are perturbed when their constituent

[234] In some of the literature this would be referred to as a change of state.
[235] This process begins when the S.O. detection process ends (see Figure 30).
[236] In other words: if nothing changes, nothing happened. A similar notion is quoted in [359].

boids see each other and maneuver (the red flock turns much more dramatically than blue). The initial boid locations are indicated using circles, and their flight paths are indicated by lines. The next step is to associate those system-level properties with a set of input properties by generating sets of symbolic regressions for each DoE simulation (in Figure 34 the example is a change in flock length). Thus, behavior association is the process of writing a system-level property as a dependent variable, which is a function of independent variables that fall into one of two categories: (1) the properties of the "other" flock, (2) the properties of a randomly selected boid from the "other" flock. These two sets of functions are depicted in Figure 34 as *f(System)* and *f(Comp.)*, respectively.[237,238] Assume, for the sake of discussion, that both functions fit the data reasonably well. In layman's terms, this means that each function is a plausible explanation for the behavior of the flock (both sets of inputs explain the behavior of the flock). However, regressions can be spurious, and so some criteria is needed to compare the two possible explanations and determine which one is the better explanation (that is the purpose of the numerical criteria in CHAPTER 4). Once the regressions are performed, the RMSE and C_T of every regression are calculated, and the function that turns out to be Pareto Optimal (lowest RMSE and C_T) is deemed to be the one more closely associated with the behavior of the system. Specifically, if the various models of *f(System)* have lower error and C_T than the models of *f(Comp.)*, then that behavior is an emergent behavior (per the criteria). If not, then it is not (by Ockham's razor, per the criteria). If such a function can be found, then it is possible to describe the behaviors and interactions of the system using a function that is not explicitly encoded into the

---

[237] Comp. is short for "component."

[238] Since the input variables are also functions of time, the dependent variable is implicitly expressed as a function of time. Time does not vanish in this approach, but the emphasis is on the initial and final states.

simulation (i.e. a behavior rule of the system has been "discovered" from the data).[239] Whether the numerical criteria can effectively support this argument is subject to the falsification of Hypothesis 2. Behavior association is implemented using the following step-by-step procedure:

1. Create a design of experiments for $\boldsymbol{n_A}$ simulations with $\boldsymbol{M}\times\boldsymbol{N}$ initial component property values, such that the components will self-organize, and the self-organized systems will interact <u>at least</u> once[240]
2. Determine the time interval $\boldsymbol{T > 0}$ and number of time-steps $\boldsymbol{n_t}$ required for the self-organized systems to interact <u>at most</u> once (usually by trial and error and/or initializing the components as a stable, organized system)[241]
3. Determine the list of $\boldsymbol{n_s}$ system-level properties, $\boldsymbol{S}$, to track (based on information from the pattern-recognition step)
4. For each simulation execution ($i \in [1, \boldsymbol{n_A}]$):
    a. Determine the time interval $\mathbf{0} < \Delta \boldsymbol{t_i} \leq \boldsymbol{T}$ at which to measure the initial and final values of the component properties (may vary per simulation)
    b. For each self-organized system $j \in [1,2]$:[242]
        i. For each system-level property ($k \in [1, \boldsymbol{n_s}]$):
            1. Use the initial and final component property values to compute the change in values of the system-level property $\Delta\boldsymbol{S_k}$ (the change in this property over time is the "behavior")
5. Filter out simulations where the systems never interact ($\boldsymbol{n_B} \leq \boldsymbol{n_A}$ simulations remaining).
6. Organize all of the values for each $\Delta\boldsymbol{S_k}$ into two large tables:[243]

---

[239] Model discovery is outside the scope of this thesis. True model discovery requires that the behavior rule be subjected to a battery of tests to ensure that the statistics used to converge to that model are appropriate, to confirm that the model extrapolates well, and to show that the model can be validated by experiment (or verified by some theory). The interested reader is referred to [348] [349] [350] [359].
[240] Each component has $\boldsymbol{M}$ properties, and there are $\boldsymbol{N}$ components in total. In tabular form, this DoE could have $\boldsymbol{M}\times\boldsymbol{N}$ columns and $\boldsymbol{A}$ rows, where each row is a different simulation.
[241] For example, one can choose $\boldsymbol{T = 1000 n_t}$ (the simulation runs for 1,000 time steps).
[242] Assume there are only two self-organized systems for simplicity.
[243] Assuming two similar interacting systems, this will produce $\boldsymbol{2\times n_s}$ tables of data (two tables for each system-level property) with $\boldsymbol{2\times n_B}$ rows per table ($\boldsymbol{n_B}$ rows of values *for each system* assuming both systems experience changes in each simulation).

a. Table 1: $\Delta \boldsymbol{S_k}$ is the output column, and the $\boldsymbol{2 \times n_s}$ input columns are the initial values of all system-level properties for both systems

b. Table 2: $\Delta \boldsymbol{S_k}$ is the output column, and the $\boldsymbol{n_s + M \times N}$ input columns are the initial values of the system-level properties for "this" system and the component-level properties for a single component of "the other" system

7. For each system-level property ($k \in [1, \boldsymbol{n_s}]$):
   a. For $\boldsymbol{G}$ regressions:[244]
      i. Using Table 1 as input to SISSO:
         1. Extract $\boldsymbol{X}$ rows of data to serve as a training set (e.g. 60% of the data).
         2. Obtain a regression for $\Delta \boldsymbol{S_k}$ as a function of system-level properties on $\boldsymbol{X}$. This is *f(System)*.
         3. Extrapolate the functions obtained for $\Delta \boldsymbol{S_k}$ onto the test data ($\boldsymbol{Y}$ rows of data, $\boldsymbol{Y = 2 \times n_B - X}$, e.g. 40% of the data).
      ii. Using Table 2 as input to SISSO:
         1. Extract $\boldsymbol{X}$ rows of data to serve as a training set.
         2. Obtain a regression for $\Delta \boldsymbol{S_k}$ as a function of component -level properties on $\boldsymbol{X}$. This is *f(Comp)*.
         3. Extrapolate the functions obtained for $\Delta \boldsymbol{S_k}$ onto the test data.
   b. From both data sets (i.e. For $\boldsymbol{2 \times G}$ functions):
      i. Filter out pathological functions (e.g. they diverge when they should be bounded, or have relative errors > 50%).
      ii. Filter out any functions that do not include properties from "the other" system (e.g. highly nonlinear functions relating $\Delta \boldsymbol{S_k}$ of system 1 to the initial values of $\boldsymbol{S_k}$ for system 1). These regressions are spurious.
   c. Compare *f(System.)* to *f(Comp.)* by applying the numerical criteria, and from those criteria, determine whether an emergent behavior has been found.

[244] Rather than interpolating on all of the simulation data to find a single function, this thesis regresses on random subsets of the simulation data in the hopes of filtering out poor quality regressions. For example, this thesis runs 30 regressions per table per system property for the boids model in CHAPTER 6.

Section 5.5.2 will describe precisely how to compare *f(System)* and *f(Comp.)*, and then how to use the result to determine that an emergent behavior has been found according to the numerical criteria.

5.3.2 Contrasting With Other Behavior Association Studies[245]

Over the course of this thesis, a few methods were found in the literature that provide something resembling a behavior association framework (to varying degrees) in addition to definitions of emergence and self-organization. This section briefly contrasts the most pertinent of them to the work in this thesis. Readers interested in these methods are strongly encouraged to read the rest of the discussion in the Appendix.

Work by Prokopenko, Boschetti, and Ryan [206] follows the formulation that self-organization precedes emergence, and that emergent properties are those of the self-organized system (although to a lesser degree than this thesis). The authors adapt an information-theoretic approach for numerically detecting self-organization.[246] They then show how a ratio of information-theoretic scalars can be used to detect emergence. Where the authors agree with this work is that self-organization can be detected numerically, as can emergence, and in doing so, the emergent properties can be associated with the self-organized system. Furthermore, the authors use complexity metrics known in computer science to perform this detection. However, there are a number of important distinctions between their work and this thesis besides the obvious limitations of a journal publication.

[245] Note that some of the studies cited here titled their work in such a way that suggests behavior association despite using a very different procedure and logic for their association process.
[246] An improvement to this thesis may involve using changes in the Excess Entropy to signal a region of interest, wherein a pattern-recognition algorithm can search for explicit structures (in the form of periodic functions described in CHAPTER 3).

(1) They do not attempt to explicitly specify necessary or sufficient conditions. It may be possible to glean some criteria from their work, but the primary purpose of their article was to make the study of emergence accessible to researchers in various fields. Their work provided inspiration for this thesis. (2) The variables considered for emergence are known in advance. Their numerical approach does not facilitate prediction of which emergent properties will arise, or how many. This thesis tests a hypothesis regarding the number of properties. However, this thesis does not examine a mechanism for deriving those properties from scratch (this will be elaborated on in CHAPTER 9). (3) The authors borrow an example of emergence from Shalizi (see references in [206]). Unfortunately, it is an example of emergence without self-organization. The authors do not explore the fact that the system (a cloud of argon gas somehow contained in a finite volume) has been artificially organized by its container (the time-averaged mean free path between atoms is constant).[247] This omission undermines the connection the authors establish between self-organization and emergence and raises the question of what causes the emergence in the first place (if not self-organization). The ontology of this thesis does not depend on that connection. In this work, the assumption is that *any organization* of interacting parts can exhibit emergence.[248] The special case of self-organized parts was selected for this thesis only to avoid making a circular argument when testing hypotheses.[249] The authors show

[247] Since this thesis is focusing on configurations that reach a form of equilibrium, time-averaging and moving averages (which are a natural fit for AR models) has been largely omitted from this discussion.
[248] Paraphrasing: emergent properties are functions of component properties and arrangement/structure, while ordinary properties are those assigned to a component without regard for structure. Therefore, ordinary properties are actually artifacts of problem simplification (the only truly ordinary objects in the universe are quarks). Emergent properties are distinguished from arbitrary structural properties by being shown to affect the properties of other objects. Here, "arbitrary" does not mean devoid of all use or meaning.
[249] When studying emergent behavior using a simulation, it is safer to study self-organized systems because their properties cannot be hard-coded into the simulation, and the mathematical formalism developed for emergent behavior detection is easily tested by extending it to another self-organized system.

how Shalizi's approach successfully concludes that most of the low-level statistical information is irrelevant, and that the property of temperature is an emergent property of the gas. However, the argument, as outlined by the authors, fails to explain why temperature has any meaning besides invoking the laws of thermodynamics *a priori*, and why the interactions of argon atoms contribute to this temperature beyond the fact that macroscopic temperature is a statistic and the behavior of the atoms is essentially random (due to the sheer number of collisions they experience). It may be the authors did not have the space or need to elaborate on those connections due to the limitations of the journal and the goal of their article (and perhaps Shalizi did in his work). Regardless of the reason, the contrast is that this thesis seeks stronger connections.[250] (4) As stated in Section 3.2,[165] the author's definitions of emergence, borrowed from Crutchfield, are incompatible with the definitions in this thesis. Though useful in biology (their examples are biological), they do not generalize in the way that weak and functional emergence do. Namely, their definitions rely on components capable of observation,[251] although it may be that the authors meant the term observation loosely. Nevertheless, in this thesis, all emergence is detected by an object external to the system (otherwise, it is undetectable, which is the primary reason engineers are so frequently surprised despite studying the components in isolation). Secondly, they cite the notion of intrinsic emergence where the components of a system somehow realize that the system itself has properties that can be capitalized on, and then

[250] Here, the work by Halley and Wrinkler provides an alternative way of thinking, wherein the argon gas may be exhibiting "simple emergence" (see [198]), but reasoning that would require adopting a distinction between self-assembly and self-organization. The interested reader is referred to [389]. For the purposes of this thesis, it is unclear how their distinction will affect the link between data compression and emergence.
[251] Observation: as opposed to brute interaction, which can still produce the same "pattern emergence" they speak of. For example, a viral infection or an amoeba consuming bacteria. Merging their example with this thesis: the emergence lies in the fact that the gazelle can hear the lion roar (one makes sound that the other detects and responds to). Whether that reaction is biological or purely mechanical is irrelevant. If a tree falls in a forest, there is emergence.

change their behavior in order to do so. This definition is incompatible with their example of the emergence of thermodynamic temperature in argon gas (temperature has no downward causation with respect to the same atoms its measurement depends upon), and does not generalize to mindless agents.

In her PhD thesis [245], Dr. Cummings developed an M&S environment for SoS simulation in order to facilitate searching for emergent behavior. Although the environment facilitates a variety of simulations, the environment itself does not perform the task of behavior association. Rather, it is up to the user to inspect the graphical output of the environment to determine which interaction and agent metrics, if any, indicate possible emergent behavior. Cummings implements the interaction metrics by Chan[252] [139] in an effort to identify emergence, but finds that "it is not just in the deviation but emergent behavior can be found in any run." [245] This appears to contradict Chan. It also appears that the definition of emergence adopted by Cummings is incompatible with the definition in this thesis (although it is compatible with Chan's). She writes (regarding a simulation of wolf predation on rabbits), "We found interesting groupings among the rabbits and wolves. This is known as emergent behavior" [245]. In this thesis, such groupings would be classified as self-organization.

In his PhD thesis, Dr. Vadim Kim [59] implemented a design space exploration approach to search for emergent behavior in simulations of complex systems. As in this thesis, Kim emphasizes weak emergence and functional emergence as the most useful concepts to build on. However, there are several points where this thesis breaks with Kim's

[252] Roughly: Chan argues that the cumulative count of agent (component) interactions is normally distributed, and deviations from Gaussian indicate emergence.

conceptual framework. For example, Kim appears to see a dichotomy between patterns exhibited by interacting components and function, writing, “analysis of patterns is not the right approach; instead, we need to examine the function of the system as a measure of the effectiveness” (see Section 2.4 of [59]). He later argues “Many researchers attempt to measure structural complexity…, but it is not clear that there is any fundamental reason to believe that structural complexity is correlated with behavioral complexity in a context-independent way” [59]. This thesis argues to the contrary in CHAPTER 3, both in that there is a context-independent link, and that context-dependence is essential to weak and functional emergence. Kim agrees that “systems engineering methods generally fail” [59] when self-organized systems exhibit emergent behavior (under the definitions in this thesis), but then scopes the topic out of his thesis saying, “This work does not attempt to measure the structural complexity. It also does not assume or imply any connection between structural complexity and behavioral complexity. The goal of this research is to focus strictly on the behavioral/dynamical complexity” [59]. Like Cummings, Kim argues that one can identify emergence in the plots of key metrics over time (although their work appears to have developed independently). While Cummings advocates for examining sharp changes in the slope of a metric, Kim argues that “Emergence is manifested by the qualitatively different probability distributions compared to non-emergent design points” [59] and specifies shifts in mean and variance as useful indicators. Ultimately, Kim claims that “Direct measures of system effectiveness are a better way of comparing system behavior rather than indirect methods such as pattern analysis” [59]. What such approaches cannot capture is the underlying direct cause for the effectiveness of the system when those underlying causes are emergent behaviors because, from the perspective of a design of experiments (DoE), those emergent behaviors are latent variables and, thus, are not included in the DoE,. Whatever sampling distribution Kim uses to search the design space may not have adequately covered the key regions of the design space that exhibit the emergent behavior, and so, there is no explicit guarantee that significant changes in mean and variance reflect the presence of emergent behavior. Furthermore, since the existence of these latent variables[373] is evidenced by the onset of self-organization (so this thesis argues), Kim's decision to disregard pattern recognition may have hindered his ability to map the DoE to relevant regions of the design space (e.g. by not factoring in initial and boundary conditions). Finally, Kim's proposed measures of emergence are intended to "enable the detection of critical transitions in behavior," [59] which is meant in a sense different from identifying novel functions in the system (i.e. no behavior association).[253] Rather, it is a quantitative shift between system functions that are known *a priori*. This may be a useful approach for known emergent behaviors, but given the absence of pattern recognition and functional graphs indicating emergence (hypergraphs, etc.), it seems that Kim's method would require supplemental diagrams to make it fully diagnostic. Whatever behavior Kim's method can discern, Kim's method is not designed to attribute that behavior to a specific configuration of the components (i.e. it assumes that all components are part of the same system without explicit regard for of their arrangement).

In her PhD thesis, Dr. Kitto [73] explored philosophical questions of modeling emergence. Kitto does not define complexity (opting instead to think of it as a scale), and while this is a compelling idea, systems science seems to require some categorization for

[253] To be clear, the phrase "not previously observed" means that over some time interval within a simulation the system does not exhibit a particular behavior because it is not organized in a way that enables that behavior to manifest (much like a light bulb will not turn on if the switch is set to "off").

the purpose of outlining the right tools to use in a particular engineering application. The main argument of her work is very interesting. She points out that models in physics are typically object-centered (think $F = ma$),[254] which is an inherently reductionist[255] perspective.[256] Stated that way, the struggles we face in identifying emergence are obvious: how can we mathematically formulate a problem using a framework that's designed for components in isolation? Certainly not easily. She proposes to resolve these issues by adopting a process-based modeling approach, where the interactions/relationships are the central focus of the model.[257] Kitto admits this will lead to a set of disparate models that do not necessarily overlap, but, as she (and later Mitchell [144]) argue, this ought to be expected. If, as she argues using numerous examples in her thesis, the emergent behavior follows naturally by reformulating the problem, then perhaps this explains why the object-centered approach in this thesis naturally produces self-organized objects. Perhaps self-organization is better thought of as the structural analog of emergent behavior. That is, self-organization is to structural decompositions what emergent behavior is to functional decompositions.[258] The fundamental elements possess nonlinear relationships that enable them to organize into a higher-level analogy of the low-level element.[259] Hence, pattern recognition techniques can be applied in a straightforward manner to the data generated by object-centered models used in this thesis to unambiguously identify a self-organized

[254] Those interested in a deep dive on the existence of objects are referred to [397].
[255] Kitto uses Reductive Analysis to denote to the perspective broadly referred to here as Reductionism.
[256] See also a recent video on quantum jumps by PBS (in particular, Schrödinger's train of thought regarding the nature of quantum objects) [403].
[257] This is compatible with my refutation of Epstein's bee hive argument in Section 2.2.3. Given the years that have passed, it is no longer possible to determine if my counter-argument was due to a subconscious internalization of Kitto's arguments.
[258] If this conjecture pans out, it calls into question whether self-organization truly precedes emergence.
[259] That is, the system possesses qualities that are equivalent to the qualities of its idealized components so long as certain conditions are met (Section 2.2.2).

object. Perhaps a similar procedure would exist using her framework. However, since there is no one-to-one mapping between form and function (in general), there remains a gap between the self-organized structure obtained in object-centered models and emergent behavior, which seems to require at least some human intervention (hence the challenges of the behavior association step discussed in reference to Figure 125, and the hypotheses in this thesis).[260] While Kitto's philosophical approach seems promising, she concludes her thesis with a warning: "Indeed, in adopting a more complex initial set of models, many more phenomena appear to be justifiably brought into the realms of physics, although at the price of so altering physics that it may no longer be considered the same field" [73]. Since this thesis is placed nearer to systems science than physics, that is not necessarily a problem, but one table flipping per thesis is more than enough.

Finally, consider again the work by Kokar et al [97]. The project itself had multiple goals, and provides several interesting questions and answers with regards to the dynamics of collective systems. Of interest here is their ontology of emergent behaviors, their global control policies resulting in emergent behaviors, and their quantitative approach to identifying a decision-maker when a system is approaching the onset of an emergent behavior. As in this thesis, the authors encountered several definitions of emergence, which they broke into three categories. Only the category of non-localized properties is compatible with the definition of weak functional emergence used in this thesis (the topics of adaptability and feedback they derive from Fromm are neglected here). Their Boids model example lists flocking as an emergent behavior, which is incompatible with the definitions in this thesis. They use the ratio of speeds to determine that the boids are

[260] See also the discussion of Bonabeau's writing in Section 3.2.

flocking (in addition to other constraints), which serves a purpose similar to the interaction equations discussed in Section 2.3 and CHAPTER 3. Of the emergent behaviors they identify in their UAV example (which are (1) trashing, (2) poorly covered facilities, (3) saturation, (4) collision, and (5) imbalanced use of resources), only the imbalanced use of resources appears to be a distinct behavior of the *swarm* of UAVs. The others are either an undesirable outcome relative to some mission requirement, or the behavior of one or more *individual* UAVs that is pathological with respect to the mission. It is unclear exactly how these behaviors were deemed emergent, so it seems that they are based on subject matter expertise (i.e. human judgment, as in the work by Moyal et al.). Later in their paper they argue that the UAVs exhibit Type IIa and IIb emergent behaviors (due to Fromm; see reference in [97]) based on whether they follow separation rules only, or separation and cohesion rules (as in the Boids model), which to the knowledge of this author, prompt the onset of self-organization or the lack thereof. In order to properly claim that the behaviors they observe are an emergent behavior, some kind of behavior association must be performed. They do not perform this step. Rather, they implement a "variety metric" (due to Holland; see Equation 33 in [97]) whose value is intended to indicate that emergence has occurred. The authors write, "As shown, when the groups are formed (undesirable state)… variety becomes flat, whereas it fluctuates when no groups are formed (desirable state)" [97]. While many of the tools they use are very interesting, their study does not follow the conceptual framework used here (though their tools could automate some of the steps described in this section). In their conclusions, the authors acknowledge that they have not yet developed a formal model of emergent behaviors, or a complete taxonomy of emergence. Therefore, the hypotheses presented here are still merited.

As shown in Figure 35, the behavior association technique (and ontology) by Cummings and Kokar et al. begins by simulating components, and then arguing for the existence of some emergent behavior. V. Kim, similarly starts from the component simulation and argues for the existence of an emergent behavior, but does not conflate self-organization with emergence. Prokopenko et al., and this thesis, begins by simulating a set of components, and arguing for the existence of a self-organized system (any behavior associated with this system immediately qualifies as weak emergence). After the system is identified, it is observed to interact with another system and from there, this thesis argues for the existence of an emergent behavior (on the basis of it being functional emergence and satisfying the conditions in CHAPTER 4).

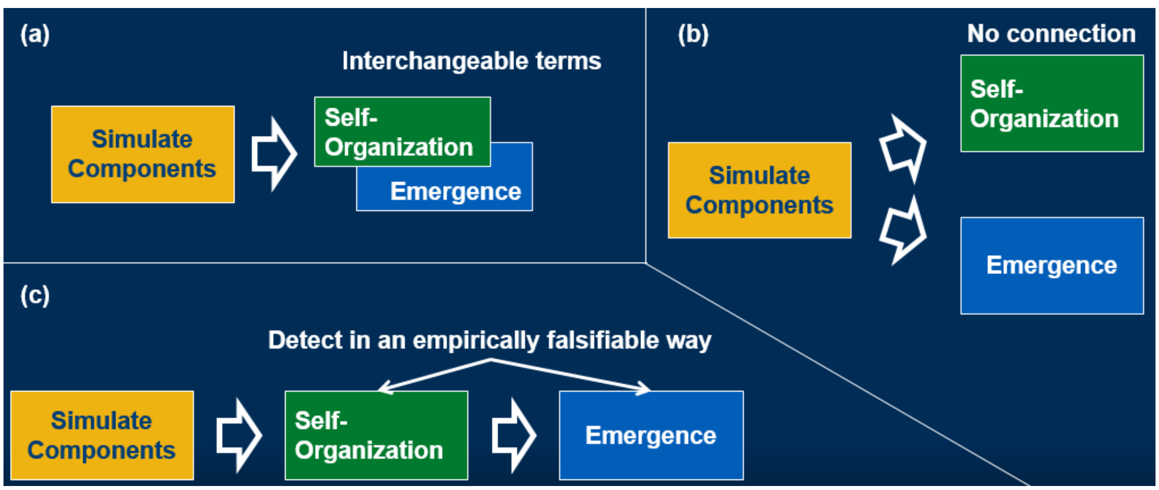


Figure 35 – Sample of relevant behavior association techniques and their implied ontologies (a) Cummings [245], and Kokar et al. [97], (b) V. Kim [59], (c) Prokopenko et al. [206], and this thesis

This thesis argues that if one can write a computable function (of any kind) that computes the numerical value of an unambiguous system-level property, the mere existence of that function (subject to certain conditions) justifies the ontological claim that said system-level property is emergent (subject to certain conditions). Without that computable function, one can only claim that the simulation shows self-organization (justified by pattern recognition) and "strange" nonlinearities resulting from low-level interaction. It may turn out (as Bedau, Huneman, and others suggest[261]) that the only way to perform that computation is by "running the simulation" and creating a large lookup table,[262] but due to the information compression and loss involved in modeling the system-level object as an idealized object, this thesis takes the optimistic stance that there exists at least some class of emergent behaviors that can be modeled using a human-interpretable mathematical expression (see CHAPTER 3) because of the measurable information those equations provide (i.e. they facilitate experiments). Thus far, scientists have written such equations when the number of components is vast (i.e. the amount of information lost/compressed it vast). Perhaps it is only in the asymptotic case that some key pieces of otherwise relevant information can be safely neglected (Section 2.2 of Balestrini-Robinson's thesis [81] provides interesting counterpoints).[263] To elucidate that, however, more studies that follow the three-phase procedure outlined in this thesis (Figure 21) would have to be performed since studies that focus on "measurable" emergence or self-organization alone do not provide enough information to make this determination. In other words, it appears that any approach to studying emergent behavior cannot proceed without some explicit focus on behavior association, however problematic it may be.[264]

---

[261] See also Kim's [58] discussion of work by Darley, Goldstein, and others.

[262] For many engineering applications this would be counterproductive due to the discrepancies between simulations and reality, which only compound when the phenomenon is nonlinear.

[263] Or, rather than asymptotically, it may be by order of magnitude. Emergence that is possible on a scale of 10 components may be meaningless at a scale of 100 components, and so forth. Perhaps the self-organization is better suited for determining scaling laws where different emergent behaviors are relevant at different scales with no or only some connection between emergent behaviors at different scales. The interested reader is referred to additional philosophical discussion in [393] [394] [395] (for mathematics, again [159] [108]).

[264] Judea Pearl might say "an explicit focus on causation." This goes without saying in engineering literature.

5.4 Stability Analysis Method for Emergence Exploitation

There is no restriction in the literature requiring that the analysis for emergent behavior exploitation for design purposes be different from that of exploitation for decision-making purposes. The presence of an "intelligent component" (systems that include people or artificial intelligence) is irrelevant, because self-organization is the basis of the pattern, not merely the rules governing the component behavior. The only difference between the engineering mindset (building a predictable system made of mindless components), and the decision-maker mindset (building a predictable system that includes intelligent components) is that in a decision-making context the apparent "rules" of behavior can change (people can learn and adapt). In a design mindset the only rules that matter are the rules of physics, chemistry, and biology.[265] Therefore, this thesis will examine two cases: (1) situations where the rules can change, (2) situations where the rules do not change.

In both situations, the task of determining which emergent behaviors to exploit and how can be performed using a Sensitivity Analysis (SA). Although there are no systematic approaches in the literature for emergent behavior exploitation, there is no need to develop such an approach from scratch (as was the case with behavior association). Sensitivity studies are common in the engineering literature for regressions of a known output from a set of known inputs. Furthermore, the conclusions one can draw from SA are generally easy to interpret and act on. There are a variety of techniques that fall under the umbrella of SA. In his textbook, "Sensitivity and Uncertainty Analysis" Physicist Dan Cacuci defines sensitivity analysis as an approach to "quantify the effects of parameter variations on calculated results" [246] where those parameters are taken to mean the constant coefficients of an equation (not to be confused with the independent variables, dependent

[265] In the broader engineering mindset the rules of ethics matter, as well as the artificially imposed rules of finance.

variables, or the form of the equation). The exploitation this thesis concerns itself with is design and behavior modification, meaning that the focus of the research is on finding dependent and independent variables. In design and behavior modification, equation parameters can be treated as satisfactory regression coefficients so long as the error of the regression is tolerable. This is very different from model discovery, where both the parameters *and* variables are of equal importance. For example, the gravitation constant in Newton's inverse square law of gravity has genuine physical significance and so must be known to the utmost precision possible.[266] Therefore, Cacuci's definition of ***sensitivity analysis*** can be modified to better suit design and behavior changes: an approach to quantify the effects of *independent variable* variations on calculated *dependent (emergent behavior) variable* results. This view of sensitivity is also common in engineering practice. Measuring this sensitivity is performed by taking derivatives of the equation and determining the magnitude of the output derivative with respect to the magnitudes of the derivatives of the input variables [226]. Note that those derivatives correspond to variables that can be directly controlled in the case of component-level properties, or indirectly controlled in the case of emergent properties. With this in mind, it is now possible to outline the four steps of an SA approach to emergent behavior exploitation:

1. Obtain a reliable regression of the Measure of Merit (MoM) that includes the emergent property or behavior in question.
2. Using the chain rule, differentiate the expression until all derivatives are expressed in terms of component-level variables.
3. Identify the component-level variables that have the largest individual (or collective) impact on the MoM, and propose an approach to affecting those variables.

[266] Showing that Newtonian gravitation is a special case of general relativity, and that the curvature of spacetime is proportional to this constant were two significant milestones in the development of the theory of gravity [329] [330]. In model discovery, the parameters can be as important as the variables themselves.

4. Simulate the proposed changes and measure the impact on the MoM.

Note that if the fit required for Step 1 is impossible, statistical tools will be needed to determine the effect an emergent behavior has on the MoM. The approach can then proceed beginning with Step 2, as will be discussed later in this section. Such an approach serves to answer Research Question 5:

> **Research Question 5:** *Once identified, how can emergent behaviors be exploited?*

In the abstract sense, the answer to this question is yes, by definition. Since the trace has been completed, the data shows that this emergent property can affect other objects, and other objects can affect this property (hence, it is exploitable). However, in physical applications, utility takes on a context-dependent meaning. To an engineer, *useful* can mean (1) reproducing the behavior in a predictable fashion and incorporating it into the intended/designed function of the system in question, or (2) avoiding the behavior altogether because it is undesirable. To a hacker, *useful* means almost the opposite: reproducing the behavior in a predictable fashion, especially those that are *un*intended functions of the system. In a competitive acquisition environment such as the context of this thesis, the goals are a combination of both: (1) fostering desirable emergent behaviors within acquired systems, (2) mitigating or eliminating undesirable emergent behaviors in acquired systems, and (3) using the emergent behaviors created by adversarial systems to their detriment. To be clear, the purpose of this step is not to prescribe a specific exploitation approach, but to outline the steps for uncovering the exploits that are available. From there, it is up to the decision maker, analyst, or engineer to select the appropriate course of action.

For exploitation analyses other than model discovery (i.e. formulating a sort of "theory" or "law" that describes all the self-organized object's interactions with the world around it), it is possible to use the upward causation equations that constitute one form of the emergent property model to guide the bulk of the decision-making, and wherever an accurate and robust interaction equation becomes available, use the additional equation to further guide the process.[267] Suppose a variable, x, belongs to component A, and another variable, y, belongs to component B (i.e. they are the properties of these components). Components A and B self-organize into System I by some pattern in the value of $x - y$, and System I now has a property $z = f(x, y)$.[268] Now suppose that the time-rate-of-change of z is the emergent behavior, e-I. Since x, and y are known, and both A and B are low-level components that can be directly controlled, then a change in e-I can be studied without an exact interaction equation relating e-I to some other system, since e-I is essentially known from the chain rule: $\frac{dz}{dt} = \frac{\partial f}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial f}{\partial y}\frac{\partial y}{\partial t}$. The question for the engineer then becomes how to change x and y over time in the desired manner (i.e. the engineer must develop an exploit for the time-varying properties). On the other hand, as will be seen in Section 5.7 and CHAPTER 7, if the system is made of components that are not under the direct control of the engineer (such as the system adversaries in combat) then the decision of how to affect a variable is no longer purely technical (because it is not possible to directly control the adversary in the combat system the way one would control a technical design variable). It now becomes a ways-versus-means decision. Once the exploit has been identified, its impact on performance can be measured by simulating system-level interactions with and without the exploit, and comparing their performance using an appropriate MoM. If the

[267] In other words, a perfect understanding of the emergent property is not necessarily needed (so long as there is a good reason for calling it emergent). Sometimes it is more important to find a way to influence it.
[268] z can be the position of the system (defined as the center of gravity of its components), a shape parameter such as length or volume, a statistic (recall that thermodynamic temperature is an average value), etc.

MoMs are totally insensitive to the exploit, this suggests a new MoM may be needed (or that the exploit is truly benign). This process is depicted in Figure 36. Therefore, this approach to exploitation analysis ties in directly with setting requirements for design, as well as the analysis needed for an effective CBA, in addition to standard engineering processes. Once the variables of interest are known and the goals of the design project can be identified, the problem becomes a regular engineering problem.

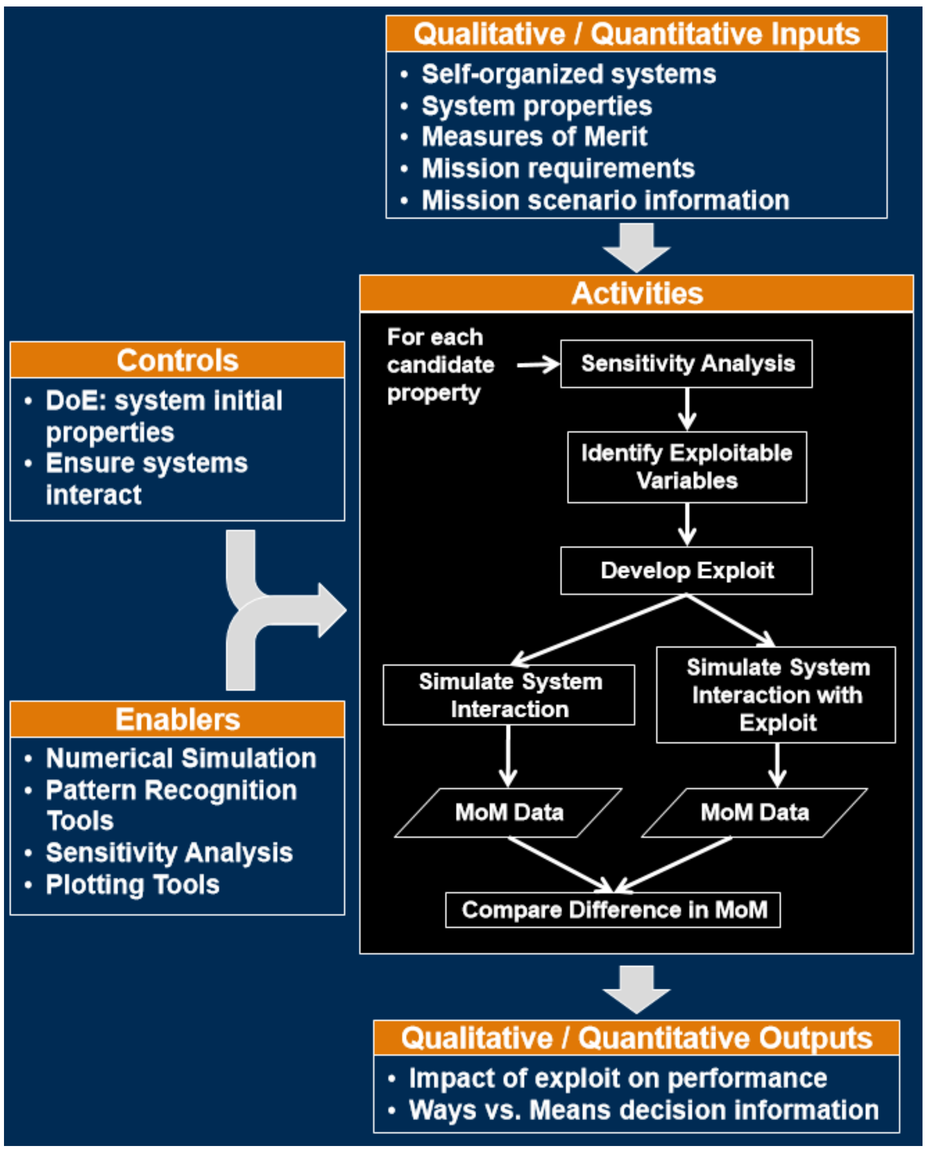


Figure 36 – Input-Process-Output diagram for exploitation analysis step[220]

The steps for implementing a generic exploitation analysis are as follows:

1. Begin with a list of $\boldsymbol{n_s}$ system-level properties, $\boldsymbol{S}$, that may be exploitable, and their equations (interaction equations from the behavior association step, or upward causation equations from the pattern recognition step).
2. Identify MoMs that are relevant for the mission in which these systems participate.
3. For each system-level property ($S_k, k \in [1, \boldsymbol{n_s}]$):
   a. Take the derivative of $\boldsymbol{S_k}$ with respect to time, applying the chain rule as needed, until the equation is expressed in terms of component-level behaviors.
      i. If there are time-varying properties present in the equation that have not been included in simulations, consider re-running the simulations DoE with these variables included (depending on available time and resources).
   b. Identify the variables, which correspond to components that can be directly controlled, and those that must be indirectly influenced (collect these variables into the set $\boldsymbol{X}$).
      i. In both cases, determine which behaviors can be affected via design changes, and which can be affected via rule changes (ways vs. means).
      ii. Determine the appropriate design change or rule change for each variable (i.e. design the exploit). Within the current method, this step must be performed manually.
   c. In the case of design changes, create a DoE with $\boldsymbol{n_A}$ rows that varies the value of variables in $\boldsymbol{X}$.[269]
      i. Include one row where the variables as unchanged in order to serve as a baseline for MoM comparisons.
   d. In the case of rule changes, create a generalization of the simulations used for the baseline rule set.
      i. The rules for the components that do not exploit the emergent behavior must be a subset of the rules for the components that do exploit the

[269] The DoE and $\boldsymbol{n_A}$ depend on the MoM being evaluated. The sample size needed to properly estimate the mean value of the MoM, for example, may be large. This is determined case-by-case.

emergent behavior, otherwise the baseline simulation cannot serve as a control case.

ii. Ensure that the code for the non-exploit rules "can be reached" by the simulation of the exploit rules, so that the baseline simulation can serve as a control case. That is, the components in the exploit-rules simulation must at least occasionally fall back onto the "default" non-exploit rules.

4. Run $\boldsymbol{n_s}$ simulations of each row in the DoE created for each system-level property design change, and measure the values of the MoM for each simulation (the size of $\boldsymbol{n_s}$ varies as some MoM statistics may require large sample sizes to converge).
5. Run $\boldsymbol{n_s}$ simulations of each component rule-set-change, and measure the values of the MoM for each simulation.
6. Compare the changes to MoM statistics that result from design changes as well as component rule changes (the decision-maker then selects the appropriate course of action).
   a. If the MoMs are completely insensitive to a design change or rule change, but the decision-maker recognizes than an exploit has been achieved, then a possible hack has been identified (an unanticipated, possibly negative, exploitable behavior).
      i. Additional MoMs are needed to properly measure the significance of this new functionality.
   b. If the MoMs are sensitive to a design change or rule change, additional simulations may be needed to determine how to encourage or avoid this new functionality. Additional simulations can provide the following information:
      i. The numerical bounds in which the designed variables cause or inhibit the emergent behavior (this requires a separate DoE, and is tantamount to a feasibility study).
      ii. The numerical bounds of the design variables that enable the rule change to be effective.
      iii. Additional rule changes that increase or decrease the likelihood of the rule-based exploitation from occurring.

A final question on this topic is the extent to which data from the behavior association step is needed for exploitation analysis. That is not straightforward to test, but if self-organization precedes emergence, then disrupting the self-organized object is equivalent to eliminating its emergent behaviors. In some applications, this may be good enough. Furthermore, under the definitions in this thesis, deliberately targeting the property of a self-organized object would immediately cause it to qualify as an emergent property (both weak and functional). This topic will be revisited in CHAPTER 7-CHAPTER 8.

5.5 Hypothesis Testing Procedure

As stated in Section 1.7, there is no mathematical approach for emergent behavior exploitation. An outline of the research presented thus far is summarized in Figure 37.

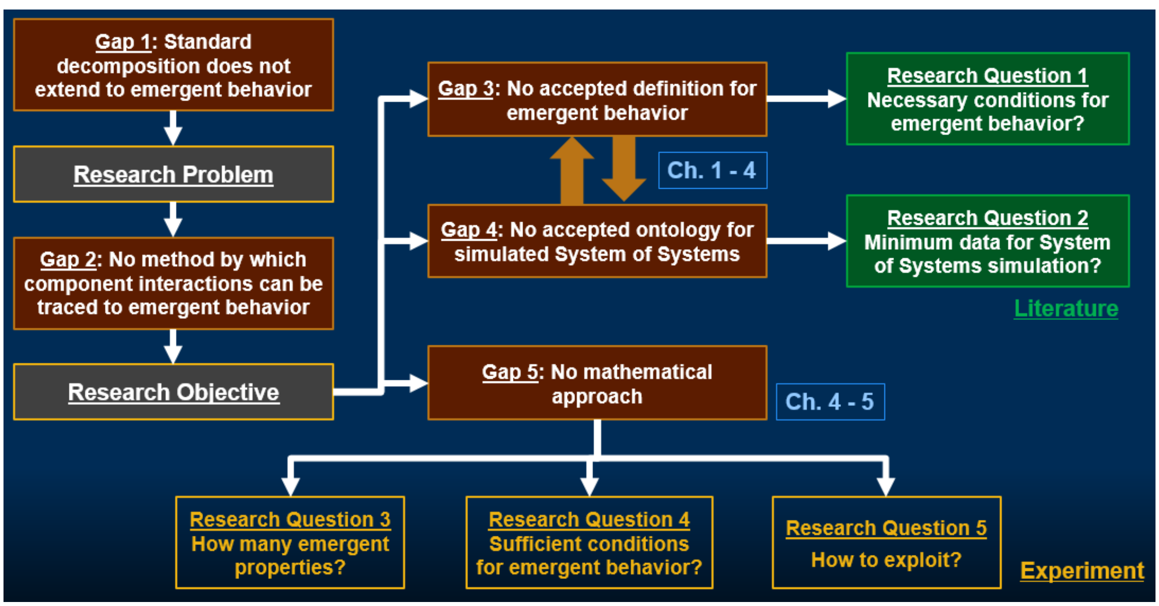


Figure 37 – Research Outline and Method for Answering Research Questions

Due to the substantial volume of work that has been published on the subject, the questions pertaining to terminology and ontology were answered from the literature. Although the mathematical approach was also derived largely from tools available in the literature, it is impossible to determine the efficacy of this method without experimentation, because there is no way to determine that the *qualitative* predictions made by the mathematical approach are valid apart from testing them. Each of the questions that follows from the mathematical approach leads to one hypothesis that must be tested empirically. The results of those experiments will be the topics of CHAPTER 6 - CHAPTER 8.

5.5.1 Hypothesis 1 Testing

Falsifying Hypothesis 1 will proceed in a straightforward manner: the total number of emergent properties identified using the numerical criteria in Section 4.3.2 will be compared against the maximum number of emergent behaviors predicted by the equation in Hypothesis 1. If the number of emergent properties found by the numerical criteria exceeds the upper bound given by Hypothesis 1, then Hypothesis 1 is falsified. For clarity, the steps of the experiment are:

1. Perform behavior association step to obtain list of $\boldsymbol{N}$ emergent behaviors belonging to system $\boldsymbol{S}$, according to numerical criteria.
2. Identify the equations that govern the time evolution of the properties of the components of $\boldsymbol{S}$ (these were explicitly coded into the simulation), and calculate the sum, $\boldsymbol{C_S(M_0)}$, of the dependent variables in this system of equations.
3. Identify the pattern equations for the components of $\boldsymbol{S}$, based on the self-organized structure of $\boldsymbol{S}$ (this is a new system of equations not explicitly coded into the simulation, obtained by the pattern recognition step).
    a. Substitute this new system of equations into the old system of equations, and eliminate any redundant variables.

b. On the simplified set of equations, calculate the sum, $\boldsymbol{C_S(M_R)}$, of the dependent variables contained in that system of equations.

4. Obtain the maximum set of emergent behaviors from the equation in Hypothesis 1: $\boldsymbol{N_{max} = C_s(M_0) - C_s(M_R) + 1}$
5. To falsify Hypothesis 1, simply find that $\boldsymbol{N} > \boldsymbol{N_{max}}$.

For example, suppose that two components (identified using the labels 1, and 2) in a simulation self-organize. At any point in the simulation, according to the rules of the simulation their x-positions are given by,

$$x_1 = V_1 t + 5.86$$

$$x_2 = V_2 t - 3.14$$

where t is time, V is the component velocity, and the constants are purely notional. Their velocities are also given by rules that can also be represented by some function in time,

$$V_1 = f(t)$$

$$V_2 = g(t)$$

The precise form of the functions, f and g, is irrelevant. Since the velocities are functions of time, they are dependent variables, as are the position variables. Then, suppose, that the as a result of self-organization, the pattern recognition step finds that,

$$x_2 - x_1 = 0.86\sin(2t) + 1.12\sin(t) - 8.4$$

Again, the coefficients do not matter. This pattern means that the original set of equations can be simplified to obtain,

$$x_1 = V_1 t + 5.86$$

$$x_2 = x_1 + 0.86\sin(2t) + 1.12\sin(t) - 8.4$$

The velocity information for the second component is redundant, and can be eliminated from the system of equations by substitution. This is the "data compression" that results from pattern recognition. In this case, one dependent variable has been eliminated from the system of equations, and the simplified set of equations is valid so long as the self-organized object persists.[270] The original system of equations had 4 independent variables (two positions, two velocities), while the new system of equations has 3 independent variables. Therefore, $N_{max} = 4 - 3 + 1 = 2$. If three or more emergent behaviors are found for the self-organized system, Hypothesis 1 is falsified. Since this mathematical approach claims that the maximum number of emergent behaviors for all self-organized systems follows this relationship, and there is no quantitative evidence in the literature to contradict that relationship (nor is there an alternative relationship in the literature), an experiment is needed to confirm that it is valid.

For the purposes of this thesis, no alternative will be explored, but the criteria raise an important question that should be considered in future work. Should the definition of self-organized system be restricted to the smallest possible representative object for the given interaction equations? For example, if a boid forms a line, the smallest system is a

[270] As the system is perturbed, the coefficients or form of the periodic relationship may vary, but as long as the system preserves its structure, there will exist a periodic function relating the two variables.

pair of boids. Should a long line be treated as a single system, or a system of systems made up of 2-boid line segments? If this is the case, then the self-organization of the long line would have to be expressed in terms of equations that use line-segment properties rather than boid properties. There is precedent for this in physics where materials are often studied in terms of the properties of their periodic unit cells. In some cases, the extension is painless, but there exist cases where there are multiple ways to examine the same structure due to its symmetries. The inequality in Hypothesis 1 does not take these extensions into account because it is supposed that this rule generalizes for all self-organized objects. The results in the chapters that follow will provide guidance for future work.

5.5.2 Hypothesis 2 Testing

Since Hypothesis 2 claims that the numerical criteria are sufficient conditions, there are two ways to falsify Hypothesis 2:

1. Checking for false positives (contradictions in definition):
   a. Find properties and behaviors that do not qualify as emergent behaviors (weak + functional) but do satisfy the criteria.
   b. Find properties and behaviors that do qualify as emergent behaviors (weak + functional) yet do not satisfy the criteria.
2. Checking that the conditions listed in the numerical criteria are complete.

Since the numerical criteria use quantifiable information to make a qualitative claim (that an emergent behavior has been found), and there is no alternative in the literature to categorically contradict it, the prediction made by the numerical criteria must be tested empirically. Care must be taken to perform these falsifications (see the Appendix for additional discussion regarding these tests). Recall that emergent behavior is a combination of weak emergence and functional emergence (borrowing from Bonabeau and Steele) as defined in Section 1.7 and CHAPTER 3, respectively:

> **Weak Emergence**: *any system-level behavior generated by component-level behaviors that can only be observed by running the simulation (or direct empirical observation).*
>
> **Functional Emergence**: *a new system-level behavior achieved indirectly by the interaction of the system's components.*

The **Criteria for Identification of Emergent Behavior from Numerical Data** from CHAPTER 4 was introduced to complete the connection between the above qualitative definitions and the quantitative data. They are the quantitative version of the qualitative definition (also in CHAPTER 4):

> **Pragmatic Definition of Emergence**: *any system-level property that can be shown to participate in an interaction (subject to restrictions).*

Hypothesis 2 is that these criteria (and hence the pragmatic definition) are sufficient conditions for weak and functional emergent behavior. Going forward weak and functional emergent behavior will be referred to simply as ***emergent behavior***.

During the behavior association step, once SISSO has generated an equation that models the interaction between two systems in a simulation, that equation will have an error (related to the data set the model was trained on) and a computational complexity. Since many simulations are being run, SISSO will generate a family of models associated with several batches of training data randomly sampled from a master data set containing all the data generated in every relevant simulation.[271] Their errors and complexity will be plotted on figures such as Figure 38.

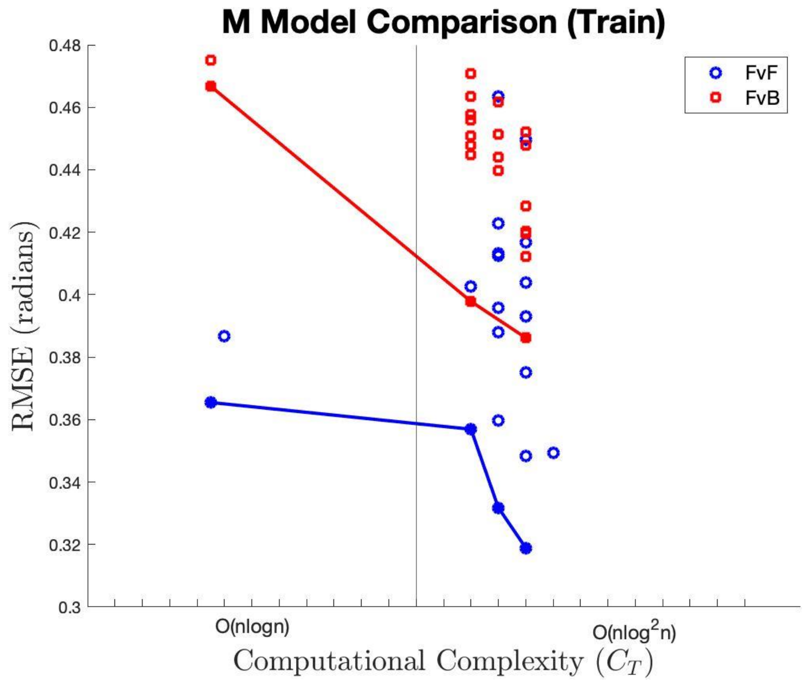


Figure 38 – Sample "point cloud" of slope, "M," and Pareto optimal set of models

Figure 38 shows the data for two sets of models of flock slope, and how that slope changes with respect to some input variables. The blue points (FvF) represent cases where the model input variables are flock-level properties. Therefore, FvF is a flock-level property expressed as a function of the "other" flock's flock-level properties (i.e. a flock-flock interaction). The red points (FvB) represent models where the input variables are boid-level properties, as though a single "other" boid was affecting the flock-level property (i.e. flock-boid interaction).[272] The distribution of the data provides two pieces of information. First, the cloud of points indicates whether FvF models tend to be more accurate and

[271] Relevant: e.g. flock interaction simulations data will only contain cases where the flocks did not split up.
[272] In order to generate the flock-boid interaction data, the boid from the "other" flock is selected randomly.

simpler than FvB models. In this case, there is a clear downward shift in the mean of the points, indicating that the models tend to be simpler and more accurate. Second, the solid lines indicate a Pareto Front drawn through using the points that are simultaneously the most accurate and the simplest (clearly, this Pareto Front is approximate). This Pareto Front will be used to test Hypothesis 2 under the strong definition of association discussed in Section 4.3.4. Note that slope is the case of a variable that boids cannot have[273] (assuming only well-defined quantities are permitted), and so it is guaranteed to be distinct. This plot shows that the FvF models dominate the FvB models, and so the flocks are more closely associated with each other. This plot shows that slope is an emergent behavior *according to the numerical criteria*. In the case of this plot, one could argue that the linear speed-up *might* undermine the dominance of the FvF Pareto front.[274] The only way to determine that it does not would be to manually inspect each equation. For example if one equation had a single multiplication, while the other had a single division, the linear speedup theorem would apply because it would be possible to design a machine that could perform division faster than multiplication by some constant factor. If all the FvB points in this graph where inside the $O(n^2 \log n)$ bin, however, the linear speed-up theorem would not apply.

More important than the linear speed-up theorem, however, is the sensitivity of the model error to extrapolation. Therefore, two plots will be generated for each model: one for the error on the training data, and another for the error on the extrapolation data (usually referred to as the test data [247]). The worse of the two outcomes will be taken as the true outcome for the sake of hypothesis testing (both plots will be provided so that the reader

[273] Not all properties are so easy to classify as distinctly high-level.
[274] If, for example, all the models in the left bin contain only multiplication, then it does not apply.

can follow the discussion in the results). Besides that, the size of the statistical sample is always important. Note that under the settings used in this thesis, SISSO will generate three different models per simulation time series (in increasing complexity). The number of time series is determined from the overall DoE size after filtering out cases where self-organization failed to persist throughout the relevant timeframe.

Figure 38 is only one figure for one system-level variable. In order to critically examine the results produced by the numerical criteria for a simulation, the criteria must be applied over several time intervals in each time series to see how the results vary based on the user-specified time interval for each variable (see Figure 32).

Table 3 – Notional comparison of property Pareto optimality across time intervals

Interval / Property	Stable / Independent (×,-)	Interaction (×,-,+)	Re-stabilization (×,-)	Interact / Re-stab. (×,-,+)	Full Time Interval (×,-,+)
Px_F	-	×	+	+	+
Py_F	×	+	-	-	×

The result of doing so will be summarized in a table resembling Table 3. The question this table seeks to answer is "can the criteria be fooled by indicating system-level interactions when there are none?" (i.e. a false positive) with an emphasis on the properties of self-organized systems. In this example, the rows correspond to the final x and y coordinate of position of the flock at the end of a given time interval. Note that the labels in Figure 32 correspond to the column labels in Table 8. In this table, a green "+" indicates that the numerical criteria were satisfied. A red "-" means that the numerical criteria are not satisfied in a weak sense (the Pareto fronts are non-dominating because they intersect).[275] A red “×” means that the numerical criteria were not satisfied in the strong sense (here, the boid-level interaction model Pareto front strongly dominates the flock-level Pareto front). Both “×” and “-” mean that the numerical criteria indicate no direct interaction, while a green “+” means they indicate a direct interaction.

Table 3 can be read column-wise or row-wise. To read column-wise, note that each column label contains the list of symbols that this author expects to see in parenthesis. For example, the Re-stabilization column is expected to contain only “×” and “-“ because there are no interactions during this time interval. The same results are expected for the Stable / Independent column. Both are testing against false interaction detection (a false positive), and so a result contradicting expectations would falsify Hypothesis 2 (the numerical criteria are not sufficient, but they may still be necessary). Row-wise, one expectation that seems plausible is that if the Interaction column exhibits a green “+” then the symbol in the Interact / Restab. column should be at least “+” if not more red,[276] and then the Full Time Interval column should continue the trend. The expectation is so because each of those time intervals is longer than the previous, beginning with the interaction time period and extending to the end of the simulation. This would support Hypothesis 2 because the emergent behavior should be most obvious over the span of the interaction, and will perhaps get obfuscated after the flock re-stabilizes. Deviations from this expectation are not necessarily conclusive, but demand an explanation. To summarize this discussion, the

[275] Another way of thinking about this (rather than using the intersection of two fronts): if all model results were collected onto one plot, and a single Pareto Front were drawn for all points, this Pareto Front would include points from both flock-boid and flock-flock interaction models.

[276] If the two Pareto fronts were gradually translating across the plot in opposite directions, the trend would be monotonic + → - → × (or the reverse).

steps for performing the experiment to test Hypothesis 2 are listed below (assuming the strict set of criteria for Hypothesis 2 from CHAPTER 4):

1. Perform the behavior association step on a given system property involved in a system-level interaction, and in particular:
    a. Using SISSO, obtain two sets of equations from the simulation time series data: (i.) interaction equations indicating that the property of system 1 is a function of the properties of system 2, which are the *f(System)* equations, (ii.) interaction equations indicating that the property of system 1 is a function of the properties of *a randomly selected component in* system 2, which are the *f(Comp.)* equations.
    b. Filter out pathological equations (poor extrapolation, etc.).
    c. Filter out equations whose variables come entirely from one system.
2. Calculate the error (RMSE) and time complexity (C_T) for all remaining equations.
3. Plot the RMSE and C_T of every equation obtained by SISSO and find the Pareto Optimal set of equations for the *f(System)* and *f(Comp.)* equations separately.
4. According to the numerical criteria, if the Pareto Front for *f(System)* dominates the Pareto Front for *f(Comp.)*, then that system-level property/behavior is an emergent property/behavior.
5. <u>False Positive Test</u>: Identify a time series that the numerical criteria finds to be an emergent behavior despite violating some aspect of the definitions provided (weak/functional emergence). For example: if no system-level interaction has taken place, then there is no evidence that the property is functional, and so the numerical criteria should not find that it is an emergent behavior.

6. Completeness Test: Identify conflicting results obtained by the numerical criteria that cannot be resolved without introducing additional criteria.

The tests will be performed using results presented in the format of Figure 38 and Table 3.

5.6 Boids Study

As stated in Section 3.3, the Boids model is an example of self-organization. Unfortunately, the properties of a flock are too abstract to permit a meaningful discussion of emergence exploitation.[277] Since the Boids model can be used to illustrate the first two steps of the overall method (Pattern Recognition and Behavior Association)[278] it will be used to test Hypotheses 1 and 2. The simulation's parameters can be broken into two categories: (1) simulation, environment, and boid properties that are constant throughout the simulation, (2) boid properties that vary over time during the simulation. The former category determines whether self-organization is feasible within the simulation, and the latter contains the variables that directly represent the time-varying behavior of the boids, and thus, the self-organization. The properties that are constant include:

Table 4 – Flocking Vee simulation and constant boid properties

Number of boids	Vision Distance	Vision Cone
Obstruction Cone	Base Speed	Speed Change Factor
Updraft Distance	Too Close	Max Turn
Length of Map	Width of Map	Space Boundary Condition

[277] The term 'exploitation' connotes an objective, which implies an optimization problem. Although goal-directedness has been associated with emergence in the literature [215] [403] [200], goal-directedness in the Boids model is too great a stretch for this thesis.

[278] An opportunity available for future is to apply the numerical criteria to the emergent properties of flocks/herds identified in [214] to see if the properties identified in that Master's thesis are consistent with the approach used here.

Most of these properties are self-explanatory. Readers interested in details not covered here are referred to the simulation's documentation [209]. In this thesis, the map in which the boids fly has periodic boundary conditions (it is a torus), and so additional care had to be taken when measuring the relative positions of boids. The boid properties that vary over time are:

Table 5 – Time-varying boid properties

Position	Speed	Heading	Boids within Vision Cone

Since flocks, like all self-organized entities, are only stable under certain conditions, the success of the experiments depends on the ability to study flocks without destroying them. Unfortunately, due to the sheer number of variables involved, generating a convex hull characterizing the set of all parameters that produce stable flocks is computationally prohibitive.

Note that the coordinate system used in Netlogo is such that the reference angle (0 degrees) is on the upward vertical axis, and positive angles are measured clockwise from there [248]. Therefore, a heading of zero corresponds to a boid flying straight up. The simulation tracks which boids see which other boids using their unique in-simulation numerical identifiers. This information is used to distinguish a meaningful self-organized flock from an accidental linear distribution of boids.[230] The speed of the boids can never go below the base speed. Also note that although the speed of the boids has no enforced upper bound in the simulation, it is generally impossible to diverge to infinity because the behavior rules prevent it. In terms of rules, each boid is always executing one of the following five rules:

- Default behavior: fly at constant heading and speed equal to base speed
- If the closest visible boid is further away than the updraft distance, then fly towards that boid and accelerate
- (else) If there are any boids within the obstruction cone, then randomly turn in order to avoid those boids
- (else) If the closest boid's distance is less than the "Too Close" distance, then slow down
- (else) Match the speed and heading of the nearest visible boid

Although the rules clearly indicate that self-organization is possible, they say nothing about the various configurations the boids can attain, or what properties those flocks will have. For the purposes of testing Hypothesis 1 and 2, it is only necessary to consider linear flocks. Not only are they the simplest shape, the analysis of their properties would be the easiest to reproduce and debate.

Each simulation will have unique variables of interest. Since the interactions of interest are between flocks and some other entity, and the experiments here are specifically designed to falsify Hypothesis 1 and 2, it is not necessary to obtain a thoroughly representative set of data for all possible settings of every variable. In order to ensure that the flocks would interact, a space-filling design of experiments of 5,000 initial configurations is used. The line flocks are initialized with various separations between boids (calculated based on their vision cone angle and vision distance so that the boids all see each other), and point at each other so that some kind of interaction is guaranteed to occur within a few iterations by crossing paths. Figure 39 depicts a simplified DoE where a variety of randomly selected locations inside the vision code of a boid (red circles labelled Boid 1) selected as destinations for a second boid (Boid 2) to intersect at a pre-determined iteration. Boid 2 is then given a variety of random starting points (blue circles) from which to fly to that destination (blue dotted path). This is a small fraction of the full DoE, which contains hundreds of starting locations, hundreds of destination locations, and several flock sizes wherein the boids can cross paths within the vision cone of any boid in the flock.

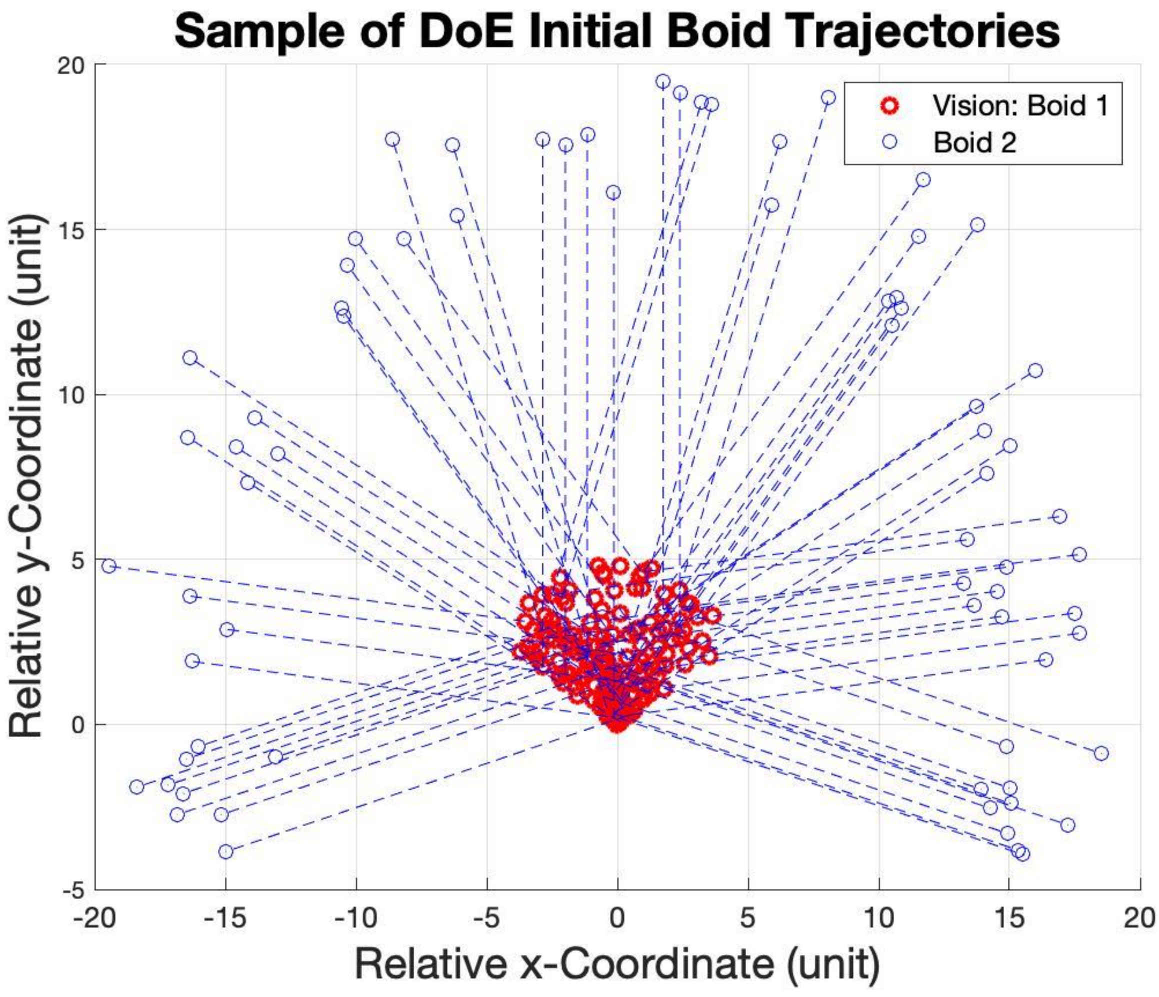


Figure 39 – Sample of DoE trajectories indicating where Boid 2 will initially intersect vision cone of Boid 1

Besides relative spacing between boids, the relative angles at which the flocks inevitably cross paths collisions are also variables in the DoE, as are the offsets between the centers of gravity at the point of intersection (if the offset is zero, then the center of gravity of each flock would overlap in the case of a perfect collision). Figure 40 depicts the full trajectories of a three-boid flock (flying vertically) crossing the path of a five-boid flock (flying down and left). In this example, the relative positions of the three-boid flock are altered dramatically as they try to retain their formation, while the five-boid flock is hardly perturbed.

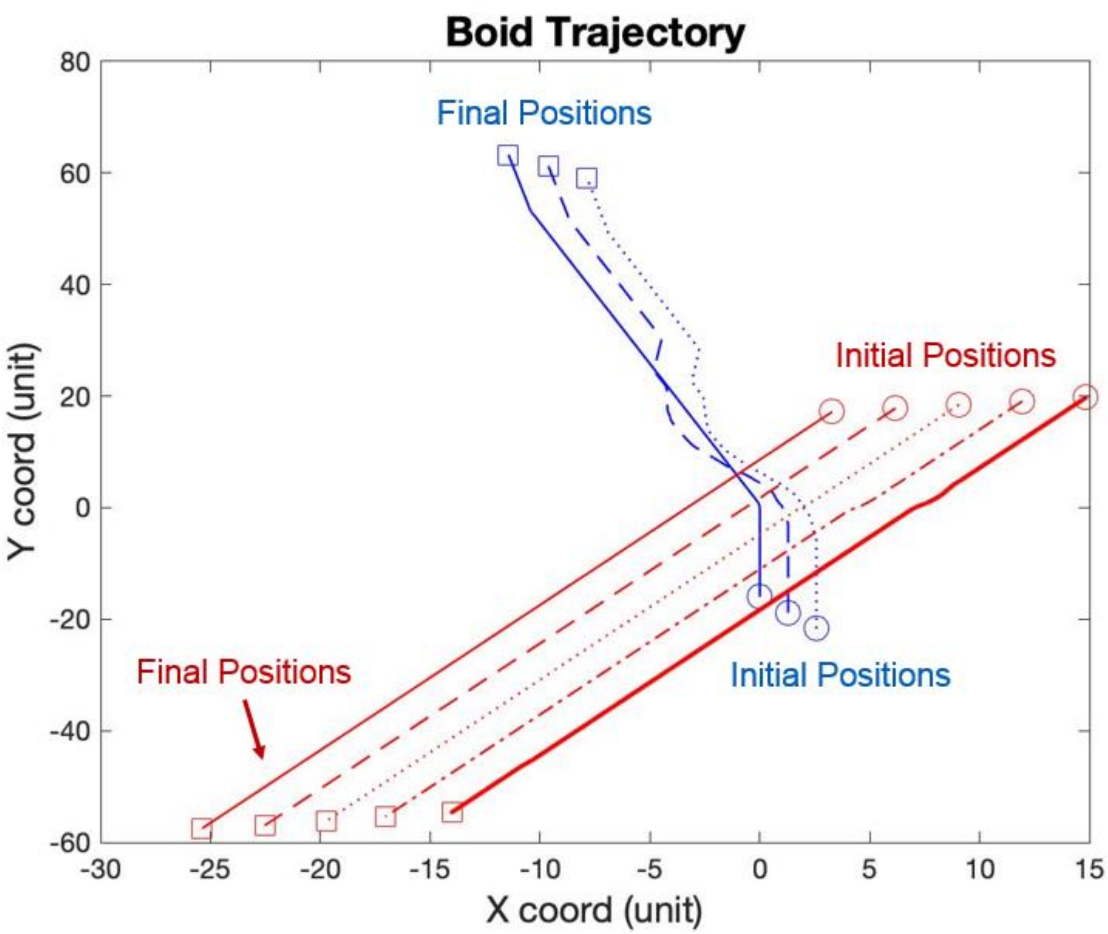


Figure 40 – Initial boid positions and intersection of two flocks

The case studies will emphasize simulations that begin with no more than two highly organized structures (the line flocks shown in Figure 17) made up of no more than five components each. The structures will be oriented so that they are guaranteed to interact once during the simulation. Since the flocks are initially self-organized, only Step 2 (Behavior Association) from Table 1 will be performed. This case will be divided into two parts: Case 1a will study the situation where a single boid crosses paths with a flock, while Case 1b will study two flocks crossing paths. After the simulations run for 400 ticks,[279] the data will be sorted into three categories: (1) flocks that re-stabilize, (2) flocks that break formation, (3) flocks that oscillate for the full simulation. Although each case can provide useful insights, the scope of this work is restricted to the flocks that re-stabilize. To confirm that the flocks re-stabilized, the positions of the boids in each flock were passed through the polyfit function in MATLAB [249], which generates a polynomial regression on the points (here, linear). If the R^2 of the regression exceeded 0.95, and if the velocities of the boids were all parallel, and if all-but-one of the boids could see each other (the lead boid does not see anyone else) during each of the last four iterations of the simulation, then the configuration was accepted as a line.

5.7 Adversarial Boids Study

Many adversarial confrontations/competitions involve some form of self-organization. World War II-style formations and dogfighting is one such example [43]. As briefly discussed in Section 1.4, LCDR Thach reviewed reports on dogfighting between Japanese and American pilots, and devised a maneuver that exploited a predictable pursuit

[279] This is slightly less than the amount of time it takes for a boid to traverse the full map once. This way, the opposing boids cross paths only once and there is ample time for flocks to re-stabilize.

scenario in air-to-air combat [250]. In this case, the self-organized system was the incoming Japanese fighter pursuing the less maneuverable American fighter from a favorable attack position (i.e. from behind the blind spot of the fighter [184]). The wingman could spot the incoming fighter [185],[280] and this gave the targeted American pilot the opportunity to act as the bait while he and his wingman maneuvered into a position that gave the wingman an advantage against the Japanese fighter.

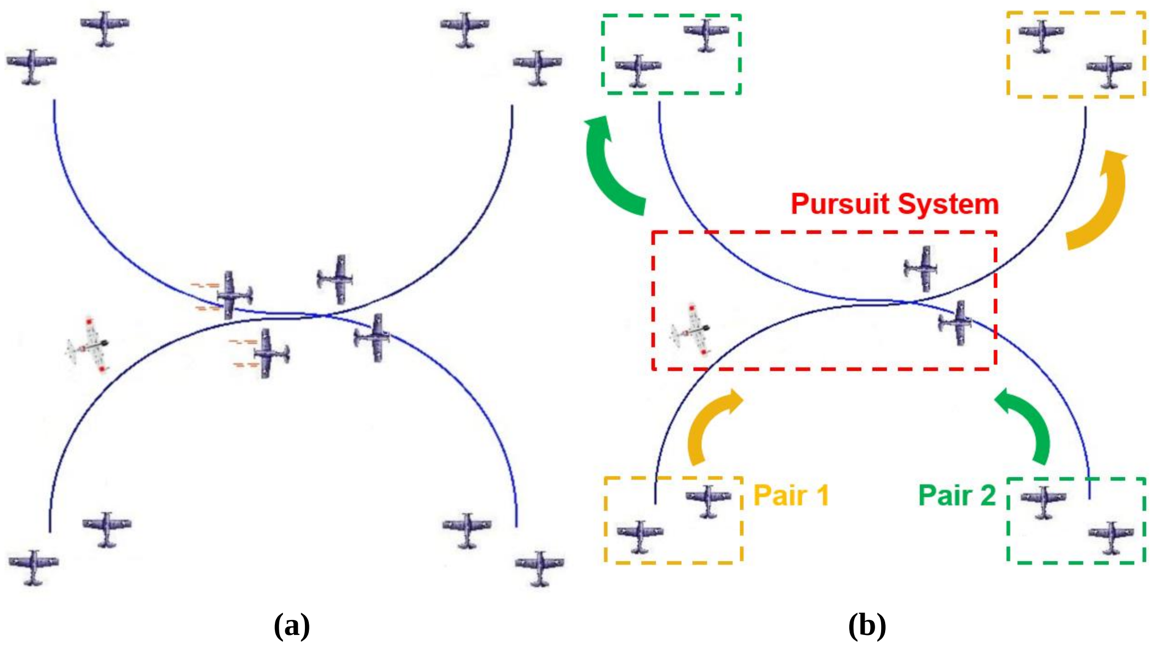


Figure 41 – Depiction of Thach Weave (a) adapted from [250], (b) annotated

The entire maneuver was predicated on the persistence of the "pursuit system" (see Figure 41), as well as the fact that it had readily identifiable and quantifiable properties. What made it successful was the fact that a very similar pursuit with very similar quantitative characteristics would inevitably come into existence multiple times each battle. Thus, the American multi-plane formations (a self-organized object) changed their usual formation

[280] Hence the formations pilots flew in, which is an example of self-organization that became codified.

to become a "bait and hook" system with each pilot assuming novel functionality in direct response to the quantifiable properties of the pursuit system. These are simple examples of systems defined by their organization, and not just their composition, as depicted in Figure 125 and Figure 126. Furthermore, these examples show how functions specific to special arrangements of components impact higher-level capabilities. A Fleet Synthesis study that cannot take such complex behaviors into account would be limited in its ability to judge between 30 year acquisition programs.

The dogfighting model used here, however, is severely limited in its realism for a number of reasons. First, NetLogo cannot perform numerical integration accurately enough for real-world engineering applications due to the sheer computational cost required to do so. Second, it is not possible in the time permitted for this thesis to develop a model with even a fraction of the sophisticated instruction provided in [185] [184]. Thus, the NetLogo model of air combat used here will aim to capture "generic dogfighting" in the same sense that Epstein's ABM's capture "generic hive-building."

Table 6 – Adversarial boids simulation and constant pilot properties

	General Parameters	
Number of red boids	Number of blue boids	Team color
Length of Map	Width of Map	Space Boundary Condition
	Vision Parameters	
Vision Distance	Obstruction Cone	Blind Spot
	Motion Parameters	
Base Speed	Red Speed Boost	Speed Change Factor
Too Close	Max Turn	Red Turn Boost
	Firing Parameters	
Max Firing Distance	Rounds per Shot	Firing Cone

Wherever convenient, the terms "pilot" and "boid" will be used to describe the basic components in the simulation. Although this model was built on top of the flocking

Vee boids model, there are significant differences between the decision-making processes of the original boids and the pilots in this simulation. As indicated in Table 6, there are two "teams" of pilots, distinguished by their colors: red and blue. There can be up to four pilots on each team. Pilots only form up with pilots of the same team. The boundary condition remains periodic, as before. The red team has "boost" parameters that will enable simulating combat against an aerodynamically superior/inferior foe. The pilots have a blind spot directly behind them to simulate the physical limitation of real-world dogfighting. Therefore, their "vision cone" is 360 degrees minus the blind spot cone, as depicted by a yellow dotted cone in Figure 42a.[281]

(a) (b)

Figure 42 – Screenshots of (a) 1v1 dogfight with attacker firing from within target's blind spot, and (2) 2v2 dogfight with red pilot countering spoiled formation

Note that the pilots can only "remember" being fired upon. This means that, during the simulation, if a visible enemy pilot maneuvers into their blind spot, the pilot will immediately "forget" that it is being chased until the attacker fires on it. Once fired upon, the pilot will have an internal timer set to 10 iterations, over which it will evade the oncoming attacker. Once the timer reaches zero, the pilot will forget that it was fired upon

[281] The blind spot depicted in Figure 42a is notional. The angle used in the simulation is smaller (15°).

and proceed to make whatever decisions its behavior rules prescribe. The timer resets any time the pilot is fired upon.

Table 7 – Time-varying pilot properties

Position	Speed	Heading	Visible Pilots
Visible Friends	Visible Enemies	Pilot of Interest	Threat level
Reward level	Timer since shot		

As with the boids, the "geometric" properties of the pilots are their position, speed, and heading. They also possess the ability to survey their surroundings and assess the level of "threat" posed by their enemies as well as the level of "reward" presented by their enemies (i.e. the vulnerability of the enemy to attack).

The "pilot of interest" is any indicator used in the code such as my-target, my-wingman, and my-obstacle, all of which facilitate executing some behavior in response to a specific pilot (the pilot has its attention focused on the pilot of interest for whatever reason). There are several very important distinctions between the logic of the simulated pilots and Boyd's OODA loop. The simulated pilots cannot be overwhelmed by too many simultaneous events. The pilots cannot respond to simultaneous problems in parallel. The pilot perfectly observe all events (to the extent that their rules permit), but only act in response to one threat or one reward per iteration (one pilot of interest).

Other key features are removed from the baseline simulation. The pilots do not communicate at all (this will be changed as part of the experiment discussed in Section 7.3). In the absence of communication, the formations exhibited by the simulation largely lose their meaning, and only affect the probability of success in a purely meaningless, mechanistic sense. Other features include: The simulation space is two dimensional. All shots land or miss based on a probability distribution that decreases with distance, and they hit instantaneously (no need to "lead the target"). This also eliminates the various tactical considerations one would have to make based on the munitions, sensors, and jamming technologies available to the pilot [43] [185] [184]. Collisions and damage do not destroy pilots, although the rules make collisions very unlikely. The two teams simply accumulate a score of shots fired, shots landed, and shots received. The pilots are a "point" rather than a two-dimensional aircraft, so collisions only occur when their positions exactly overlap.

Some key dogfighting features are persevered (albeit notionally). The rules are set up so that a pilot can force its attacker to break off pursuit by causing it to overshoot during an attack [184]. There is a mechanism by which attackers can effectively sneak up on their targets [184] [185]. Friendly pilots can intercept an attacker mid-pursuit (a precursor to the Thach weave). Multiple pilots can shepherd a single adversary in an unfavorable position for an extended period of time (another precursor to the Thach weave). All of this occurs due to a fairly simple set of rules and serial decision-making.

Although Boyd's OODA loop cannot be faithfully reproduced in this simulation, the code is organized in a manner inspired by the principal functions in the OODA loop. Each iteration, the pilots first *observe* the various pilots within their field of vision and sort them into the categories of friend or enemy. The pilots then *orient* themselves toward a particular threat (a possible collision or incoming attacker), or a possible reward (attacking an appropriate target, or entering into a formation). The various threat or reward level computations are functions of the threat/reward type and the distance between the pilot and the relevant alternative pilot (depicted in Figure 43). The threat and reward curves are set up in such a way that one behavior is more likely at a particular distance than another. For example, collision avoidance takes over when the pilots are less than four units apart (see the blue "Collision" curve in Figure 43). Note that the "Form-up" behavior is the lowest priority behavior to occur at any distance. Also note that Figure 43 contains a plot of the likelihood of landing a shot as a function of distance (the purple %-Hit curve). The pilots fire three-shot bursts in order to increase the likelihood of landing a shot (the probabilities are independent and identically distributed), and the probability of actually firing a single burst at a target is three-times the probability of hitting the target. The option of deciding to fire a shot is only triggered when the pilot sees another pilot within its firing cone (a small sector designed to enable more frequent engagement in less-than-perfect situations).

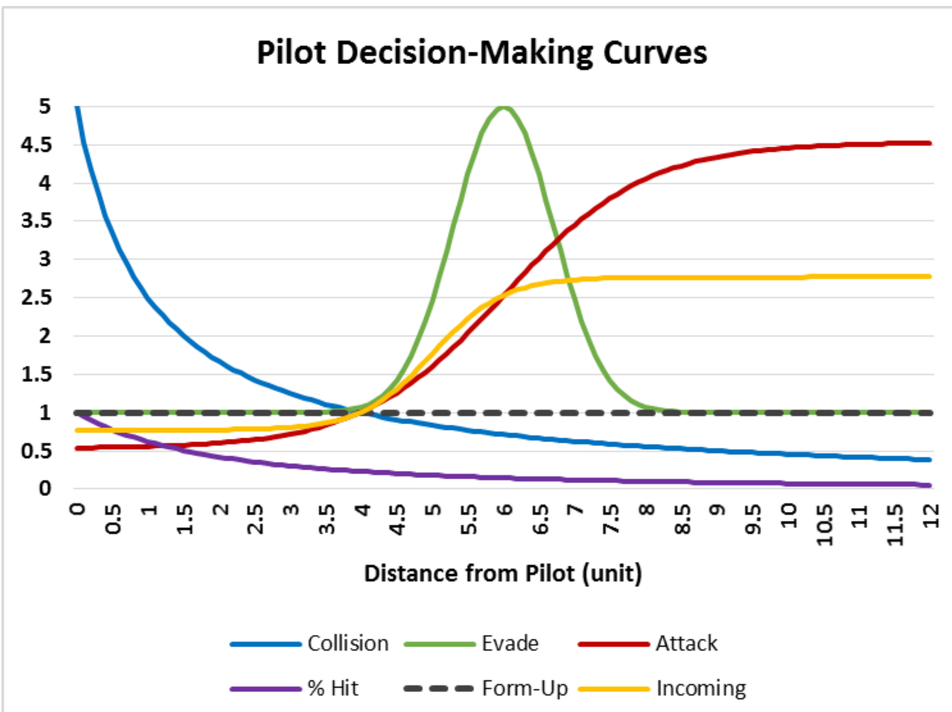


Figure 43 – Pilot decision-making threat/reward curves

The pilots then *decide* which course of action to take based on the greater of the maximum threat level and the maximum reward level (with a tie breaking in favor of threats), and then *act* (i.e. each pilot executes its decision). Although Netlogo issues the instructions sequentially (in random order each iteration), the pilots do not re-evaluate their decisions during that same iteration. Only one behavior sequence is performed per iteration. There is plenty of room for debate on how to best implement an OODA loop,[282] but that is not the primary focus of this thesis, and so is left as future work. Finally, to further clarify Figure 43, note that the yellow “Incoming” curve dictates when the pilot attempts to evade a *potential* attacker (i.e. it tries to stay out of the incoming pilot’s firing cone), whereas the green “Evade” curve dictates when the pilot evades an attacking pilot that has already at it. The red “Attack” curve dictates when the pilot selects and attacks a target. In addition to distance requirements, the pilots will prefer attacking targets they can approach from behind, but will occasionally make opportunistic attacks if they can do so without flying into the enemy’s firing cone.

This case study will focus on the first and third step in Table 1. Since the self-organized systems in this simulation will be less stable, the approach in Section 5.2 will be pushed to its limits. Rather than considering a broad set of candidate emergent behaviors, as in the boids study, this case will focus on the separation between the attacker and the target (i.e. the “length” of the pursuit system). The exploitation analysis will proceed by performing a sensitivity analysis on this length, and result in proposed design changes as well as pilot-behavior changes. The MoM for the adversarial boids study will be the ratio of shots fired at enemies divided by the shots received from an enemy. This ratio will be

[282] In real-world situations, the time interval over which people process their individual OODA loop is orders of magnitude shorter than most of the events transpiring in the battle field. Training and experience contribute to habits that streamline some of these processes which free up the person’s conscious attention to other factors, thereby enabling more sophisticated and faster decision-making.

used to compare the effectiveness of the design changes and behavior changes to the baseline pilot performance, as well as the sensitivity analysis approach overall. This will require running simulating three cases:

- The red team has superior maneuverability
- Both teams have the same maneuverability
- Red team has superior maneuverability but blue team has added behaviors to capitalize on the self-organized pursuit system

The simulations will be 6,000 iterations long in order to obtain reasonable statistics on the values of the MoM. This is 15 times longer than the time interval used in the Boids Study (Section 5.6), which means the pilots can cross the map 15 times over and have ample time for numerous engagements. This time interval was chosen because it is "long enough" to capture the dynamics of multiple engagements based on experience with the simulation. Since the pilots are initialized randomly, there is no guarantee that there will be any number of engagements in a given simulation, or that some useful statistic will converge over the course of a single, infinitely long simulation. This uncertainty will be compensated for by running a large enough sample of simulations (see Section 7.4.2).

CHAPTER 6. THE BOIDS MODEL CASE STUDY

Boids have only two interaction-dependent properties (heading and speed) in the sense that these two properties are constant until an interaction occurs. These interactions are one-way. Each boid changes its trajectory in response to seeing one or more other boids only (those other boids need not see the boid that boid that is "interacting" with them). Any semblance of avoiding a collision is coincidental, and nothing happens to the boids if their positions overlap. Finally, note that the sub-section headings in this chapter that correspond directly to specialized hypothesis testing will be labelled with "H1" for hypothesis 1, and "H2" for hypothesis 2. Data that is relevant to the hypothesis but not amenable to a straightforward test will be discussed over the course of multiple sections and summarized at the end of the chapter.

6.1 Case 1a: Boids versus Flock

Unlike the case of two interacting flocks, when a flock interacts with a single boid (as shown in Figure 44), the boid stabilizes as soon as the interaction is over. Thus, for the single boid, the interaction time interval is the full interval of interest. Since the speed of the boid only changes during the interaction, if at all, the full impact of the single boid's speed will not knowable at the end of the interaction time interval. Thus, the solitary boid only has one "useful" property for the purpose of hypothesis testing, and that is its heading (only heading permanently changes). The interaction that will be examined here is the interaction of the solitary boid with the flock. Therefore, the data used here will correspond to the time interval beginning when the solitary boid first sees any boid in the opposing flock, and will end when the solitary boid resumes constant heading and speed (the values of the properties of the flock at that instant will be used, whether it has re-stabilized or not). In other words, the output/dependent variable modeled in this section will always be the heading of the solitary boid, and the input variables will be either flock-level properties or boid-level properties. In this sense, this section will test whether the heading of the solitary boid is more sensitive to the properties of the flock, or the properties of a randomly-selected boid within the opposing flock. Cases where the solitary boid merges with the flock will be discarded.[283]

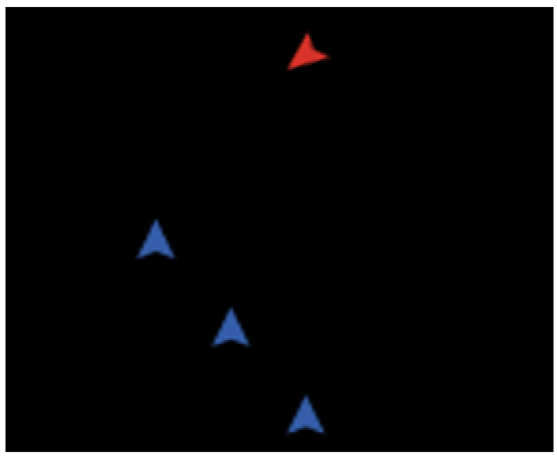
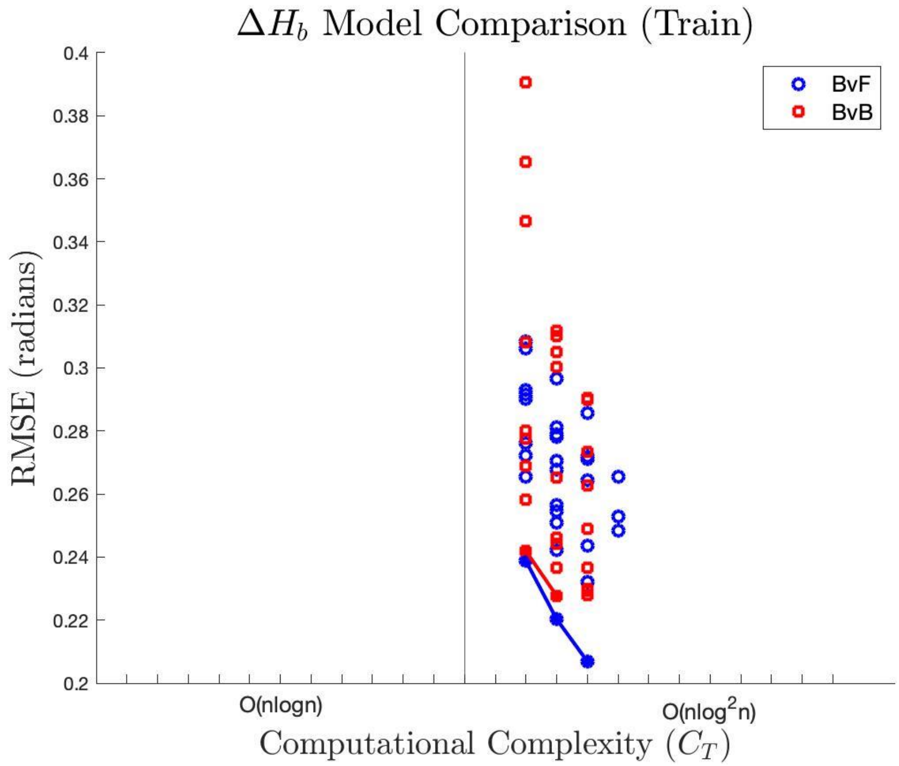


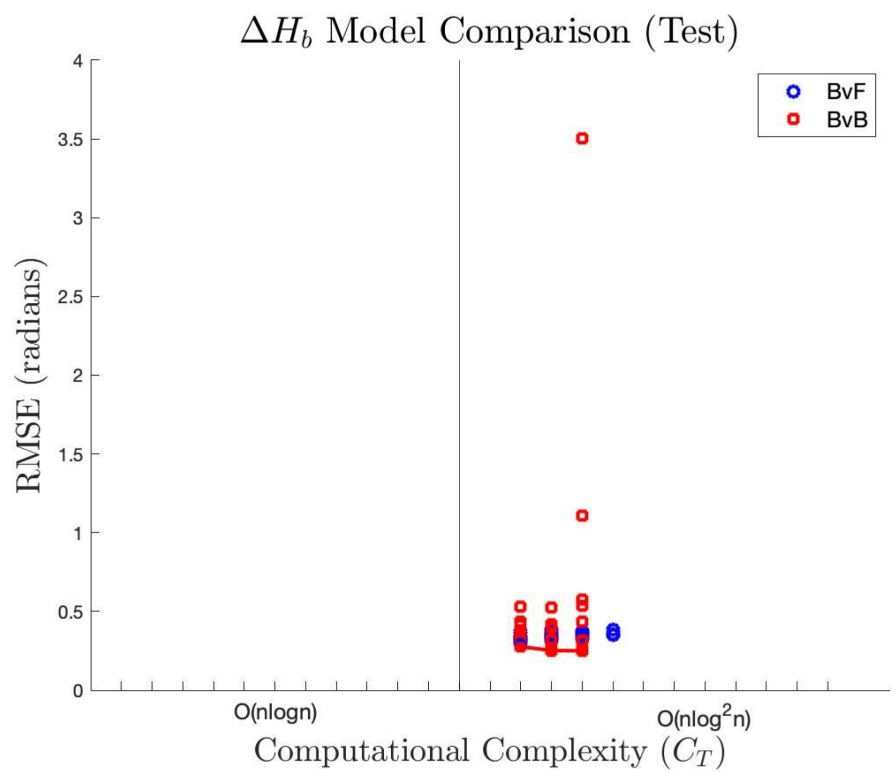
Figure 44 – Boid on collision course with flock for Case 1a study

The Pareto Fronts for the interpolation data (Figure 45a) show that the models using flock variables as inputs (BvF) are more accurate and simpler (lower C_T) than the models using boid variables as inputs (BvB). Over the extrapolation/test data (Figure 45b), however, the BvF models are just barely dominated by the BvB models. There is a lot of information to consider from these plots.

[283] The "growth" of the flock is a behavior on the border between emergent behavior and self-organization. Clarifying this concept will be left for future work.



(a)



(b)

Figure 45 – Pareto optimal boid heading models due to interaction (a) training data (b) test data

Since the dependent variable being tested here is the heading of a solitary boid (which cannot be an emergent behavior), the question here is whether or not the boid's behavior is a direct response to the properties of the flock (which can be emergent) more so than the properties of other boids. One could argue that this scenario is a form of downward causation. Based on the numerical criteria alone, and taking the more conservative result of the two figures, the BvF models are dominated by the BvB models, and so there is no indication that the flock-level properties are emergent behaviors. However, since downward causation has not been extensively studied, these results are not conclusive. Nevertheless, it is worth mentioning because the common assumption surrounding emergent behaviors (particularly those in biology) is that there exist objects that respond to groups differently than they do to individual objects at their level of abstraction. There is, however, a more compelling feature of these results to consider.

Two of the models[284] comprising the Pareto Fronts depicted in Figure 45b are,

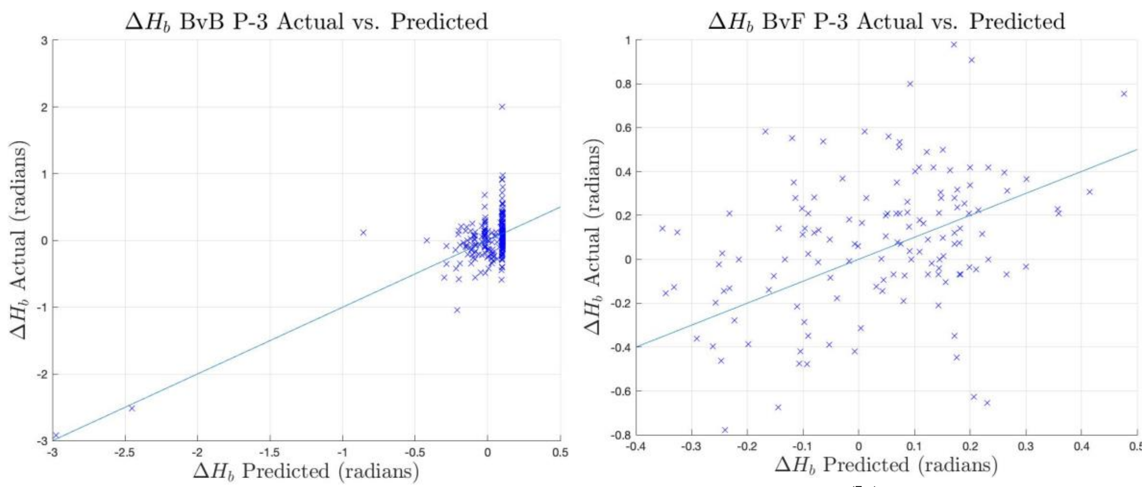
BvF: $$H_{b1,f} - H_{b1,0} = 0.147 - 8.4 \times 10^{-4} D_{F,0}^{3} \cos L_{F,0} \tag{4}$$

BvB: $$H_{b1,f} - H_{b1,0} = 0.34 - 0.313 e^{D_{b2,0} H_{b2,0}} \tag{5}$$

where D is the distance between the boid and the other object,[285] L is the length of the flock, H is the heading, the subscript F denotes the flock, the subscript 1 denotes the solitary boid, the subscript 2 denotes any boid in the flock, and, finally, the subscripts 0 and f denote initial and final, respectively. Of the two models, the BvF models is the most believable

[284] There are four in total. The third takes the form of (x-y) + |x-y| which is a recurring result and will be discussed further in Section 6.2. The fourth model (a BvB model) is not particularly informative.
[285] The position of the flock is the centroid of the positions of its constituent boids.

because it suggests that once the boid passes through the center of the flock, it no longer changes its heading, which is generally correct since the only way it does this is by flying head-on into the flock (i.e. once it passes through the middle, it can no longer see any other boids). Furthermore, it is never zero, nor ever constant (with respect to the variation of parameters in the design of experiments). On the other hand, the BvB model suggests that if the heading of the opposing boid in the flock is zero, then the change in heading of the solitary boid is always constant. This is difficult to accept since there is nothing special about a boid whose heading is zero. Furthermore, the maximum absolute errors of the BvB models are significantly higher than those of the BvF models on the interpolation data, in addition to having very poor distributions of errors.



(a) **(b)**

Figure 46 – Solitary boid heading Actual vs. Predicted plots (test data models applied to interpolation and extrapolation data) for Pareto Optimal models

Note that the results in Figure 46 are representative of all BvB and BvF models in their respective Pareto Fronts. Such results indicate that it may be necessary to incorporate additional model quality tests in order to make the current numerical criteria sufficient conditions, or at least more robust to spurious regressions.

6.2 Case 1b: Flock versus Flock

This case will focus on two self-organized systems interacting,[286] and so there will be many more attempts to falsify Hypothesis 2 using this data set. The time intervals referenced here match with those depicted in Figure 32 (the subsection headers will follow their naming convention). Sections 6.2.1 and 6.2.3 will directly test the limitations of the numerical criteria by testing whether they can produce false positive detections of interaction. Sections 6.2.2, 6.2.4, and 6.2.5 will provide the data needed to perform the aforementioned[287] critical examination of trends to determine whether the numerical criteria suffice for emergent behavior detection. Due to the volume of data and discussion, the results from each section are tabulated and summarized in Section 6.2.6.

One important distinction between the data analysis performed here and the analysis performed in Section 6.1 is that the beginning and end of the interaction time intervals for the flocks no longer coincide with the interaction time interval of the individual boids. That is because there is a difference between the time interval over which a flock interacts with a boid, and the time interval over which a flock interacts with the flock containing said boid. The time interval of the flock interactions is much longer since it lasts so long as any boid in one flock observes any boid in the opposing flock. One or more boids of the opposing flock can participate in this interaction sequentially, jointly, or sometimes not at all. For the sake of clarity, consider the following example: flock Red is interacting with flock Blue, and flock Blue contains boids X, Y, and Z. The first boid that comes within visual range of flock Red is boid X (this is the beginning of both the Red-

[286] Recall that the flocks are initialized in their pre-determined shape (see Section 5.6).
[287] Hypothesis falsification approach #3 in Section 5.5.2

Blue interaction and the Red-X interaction). Suppose flock Blue is also perturbed because a boid from Red is visible to boid X. That means that boid Y, which is not yet participating in the interaction (it only sees X), may have already begun adjusting its heading and speed in an attempt to stay in formation (to re-stabilize flock Blue). Therefore, by the time flock Red begins to interact with boid Y, it has already deviated from its original heading and speed. Therefore, the initial heading and speed of boid X is the same as the initial heading and speed of flock Blue when the interaction begins, but the initial heading and speed of boid Y will be different than that of flock Blue when it begins its interaction with flock Red. Thus, the interchangeable property fallacy described Section 5.5.2 (the second way of fooling the criteria) cannot occur for the data in Sections 6.2.2 - 6.2.4. It can occur in Section 6.2.5, and will be discussed in that section.

Finally, consider the following example of the notation that will be used in the equations in Sections 6.2.2 - 6.2.5:

$$\text{FvB:} \quad L_{F1,f} = 0.291 + L_{F1,0} - 1{,}682\left(S_{F1,0}\right)^{6}\left|H_{2,0} - H_{F1,0}\right| \tag{6}$$

In Eq. ((6) the opposing flocks are labelled $_1$ and $_2$, where the number 2 corresponds to the "other" flock. The equation is labelled FvB to indicate that it is an interaction equation between a flock and a randomly selected boid from the other flock. FvF would indicate an interaction between two flocks, and the equations would contain only flock-level variables. Variables with the subscript $_F$ are flock variables, while those without that subscript are boid variables. The $_f$ subscript means final (with respect to the time-interval in question), while the subscript $_0$ corresponds to the initial value. Therefore, Eq. ((6) is the model for the final length of a flock. The final length is given in terms of the initial length of the flock, the initial speed of that same flock, and the absolute value of the difference between the initial heading of the flock and the initial heading of a randomly selected boid from the opposing flock. Notice that these models assume that the dependent variable is an explicit function of other properties/variables (and so, it is an implicit function of time)

The notation in the following section will be slightly more descriptive in order to facilitate the narrative. Rather than flocks 1 and 2, the flocks will be labelled red and blue so that the reader can easily visualize the behavior described by the equations. Also, rather than a randomly selected boid, the equations will be based on a specific boid (again, to facilitate understanding).

6.2.1 Stable / Independent

First, consider the case where two flocks fly away from each other without interacting. If two totally independent behaviors satisfy the criteria for emergent behavior, then Hypothesis 2 is falsified because if it is not sufficient for interaction detection, then it is not sufficient for behavior association, and hence, emergent behavior identification. Since both flocks move at a constant speed and heading, it is easy to rewrite the time-varying position of one flock as a linear function of the other. Therefore, this experiment will be conducted using “pencil and paper.” For simplicity, suppose that the two opposing flocks are colored red and blue.

It must be shown that this non-emergent behavior fails to satisfy the numerical criteria, otherwise, there is a contradiction. The time-varying positions of the red flock and blue flock are given by the following equations,

$$P_{y,red} = V_{y,red}t + P_{y0,red} \tag{7}$$

$$P_{y,blue} = V_{y,blue}t + P_{y0,blue} \tag{8}$$

where, P_y, V_y, P_{y0}, are the y-position, y-velocity component, and initial y-position of the flock, respectively, and *t* is time. Dividing the red equation by the blue equation we obtain,

$$\frac{P_{y,red} - P_{y0,red}}{P_{y,blue} - P_{y0,blue}} = \frac{V_{y,red}t}{V_{y,blue}t} \tag{9}$$

Simplifying the equation, and rearranging terms,

$$P_{y,red} - P_{y0,red} = \frac{V_{y,red}}{V_{y,blue}}\left(P_{y,blue} - P_{y0,blue}\right) \tag{10}$$

$$P_{y,red} = \left(P_{y0,red} - \frac{V_{y,red}}{V_{y,blue}}P_{y0,blue}\right) + \frac{V_{y,red}}{V_{y,blue}}P_{y,blue} \tag{11}$$

When the flocks are flying away from one-another and have not interacted, *V* is constant (as are the initial conditions), and so the equation can be rewritten as,

$$P_{y,red} = c_0 + c_1 P_{y,blue} \tag{12}$$

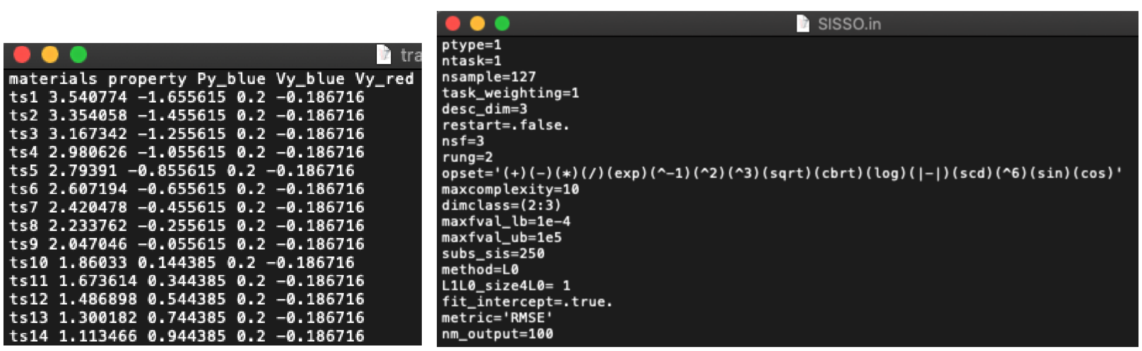
where c_0 and c_1 are constants. The same equation can be derived for any boid that is a member of the blue flock. Suppose this boid is labelled “1.” Its position is given by,

$$P_{y,1} = V_{y,1}t + P_{y0,1} \tag{13}$$

Following the procedure as before, but now using the equation for boid 1,

$$P_{y,red} = c_0 + c_1 P_{y,1} \tag{14}$$

Although the constants are different for Eq. (12) and Eq. (14), the forms of each equation are the same. Thus, they have the same time complexity, $C_T = O(n\log n)$, due to the single multiplication, and the error of both equations is the same, RMSE = 0. Using the "weak association" formulation of Hypothesis 2, since the flock properties are *as closely* associated to one-another as the red flock property is to the boid 1 property (rather than *more closely*), then the flocks do *not* directly interact using this property, and so the position of the flock is *not* an emergent property. This result is important because it shows that two totally independent properties do not qualify as emergent properties under Hypothesis 2.[288] Thus far, Hypothesis 2 is supported.



```
materials property Py_blue Vy_blue Vy_red
ts1 3.540774 -1.655615 0.2 -0.186716
ts2 3.354058 -1.455615 0.2 -0.186716
ts3 3.167342 -1.255615 0.2 -0.186716
ts4 2.980626 -1.055615 0.2 -0.186716
ts5 2.79391 -0.855615 0.2 -0.186716
ts6 2.607194 -0.655615 0.2 -0.186716
ts7 2.420478 -0.455615 0.2 -0.186716
ts8 2.233762 -0.255615 0.2 -0.186716
ts9 2.047046 -0.055615 0.2 -0.186716
ts10 1.86033 0.144385 0.2 -0.186716
ts11 1.673614 0.344385 0.2 -0.186716
ts12 1.486898 0.544385 0.2 -0.186716
ts13 1.300182 0.744385 0.2 -0.186716
ts14 1.113466 0.944385 0.2 -0.186716
```

```
ptype=1
ntask=1
nsample=127
task_weighting=1
desc_dim=3
restart=.false.
nsf=3
rung=2
opset='(+)(-)(*)(/)(exp)(^-1)(^2)(^3)(sqrt)(cbrt)(log)(|-|)(scd)(^6)(sin)(cos)'
maxcomplexity=10
dimclass=(2:3)
maxfval_lb=1e-4
maxfval_ub=1e5
subs_sis=250
method=L0
L1L0_size4L0= 1
fit_intercept=.true.
metric='RMSE'
nm_output=100
```

(a) **(b)**

Figure 47 – Analytical test of Hypothesis 2 (a) SISSO time series input training data (b) SISSO settings file

This analytical test can also be performed using SISSO with the velocities and positions as input variables (recall that the velocity is constant throughout). The interested

[288] Or perhaps this works because the equations are linear. Section 6.2.3 will test whether or not this extends to non-linear cases.

reader can verify the forthcoming calculations using the data in the screenshots of the input files provided in Figure 47 below.The result returned by SISSO, depicted in Figure 48, is the equation:

$$P_{y,red} = 1.124 - 4.6679(V_{y,red} + P_{y,blue}V_{y,blue}) \tag{15}$$

Observe that,

$$-4.6679 = {V_{y,red}}/{V_{y,blue}^2} \tag{16}$$

$$1.124 \cong -\left({V_{y,red}}/{V_{y,blue}}\right)^2 - {V_{y,red}}/{V_{y,blue}}\, P_{y0,blue} + P_{y0,red} \tag{17}$$

Substituting these equations into the solution SISSO yields,

$$P_{y,red} = -\left(\frac{V_{y,red}}{V_{y,blue}}\right)^2 - \frac{V_{y,red}}{V_{y,blue}} P_{y0,blue} + P_{y0,red} + \frac{V_{y,red}}{V_{y,blue}^2}(V_{y,red} + P_{y,blue}V_{y,blue}) \tag{18}$$

which simplifies to Eq. (12). Note that although the equation is exact, SISSO reports an RMSE of 1.217E-15, which is around the limit of machine accuracy (double precision floating point arithmetic in this case).

```
Model/descriptor for generating residual:
================================================================================
  1D descriptor (model):
Total RMSE,MaxAE:   0.000000  0.000000
@@@descriptor:
                      5:[(Vy_red+(Py_blue*Vy_blue))]
       coefficients_001:    -0.4667900000E+01
          Intercept_001:     0.1123553332E+01
         RMSE,MaxAE_001:     0.1216891444E-14    0.3552713679E-14
================================================================================
Wall-clock time (second) for this DI:            0.07
DI done!
```

Figure 48 - Analytical test of Hypothesis 2, SISSO output file (1D descriptor)

Clearly in this case, SISSO did not find the same functional form as the analytical case (prior to simplification). This has significant consequences on the naïve calculation of time complexity from SISSO outputs. The SISSO result contains two multiplications whereas the manually-derived example above only has one. Therefore, SISSO cannot necessarily be relied on to find the simplest form of an equation from a time-complexity standpoint with a naïve input file.

Despite that drawback, there are four justifications for using SISSO in this thesis. First, constant inputs are a pathological example for SISSO, and the studies in this thesis will not use constant-value input.[289] Second, the strict interaction criteria uses Pareto Fronts to compare the time complexity of equations. Although it is possible for SISSO to produce a Pareto Front of overly-complex equations, it probably will not to do this for all results.[290] Furthermore, since all analyses are proceeding through SISSO, whatever bias SISSO introduces into the complexity of the equations will be applied systematically across all results, and can be corrected with further study. Third, this uniqueness problem, wherein the most accurate model can be written multiple ways, is a fundamentally inescapable

[289] It won't necessarily crash the program, but experience suggests it was not designed for such cases.
[290] This is a judgment call.

problem, as described in Section 2.2. Therefore, even programs designed for time-series studies such as PySINDy cannot fully escape this limitation. Fourth, the result in Eq. (18) is due to SISSO overcompensating for the initial conditions. The initial condition can be subtracted out of the time-series, as shown in Figure 49,

```
materials property Py_blue Vy_blue Vy_red
ts1 0 0 0.2 -0.186716
ts2 -0.186716 0.2 0.2 -0.186716
ts3 -0.373432 0.4 0.2 -0.186716
ts4 -0.560148 0.6 0.2 -0.186716
ts5 -0.746864 0.8 0.2 -0.186716
ts6 -0.93358 1 0.2 -0.186716
ts7 -1.120296 1.2 0.2 -0.186716
ts8 -1.307012 1.4 0.2 -0.186716
ts9 -1.493728 1.6 0.2 -0.186716
ts10 -1.680444 1.8 0.2 -0.186716
ts11 -1.86716 2 0.2 -0.186716
```

Figure 49 – Input training data (minus initial conditions)

SISSO then obtains the result (see also Figure 50),

$$P_{y,red} = -0.9336P_{y,1} \tag{19}$$

The intercept is understood to be equal to zero despite the round-off error.

```
Model/descriptor for generating residual:
================================================================================
  1D descriptor (model):
Total RMSE,MaxAE:   0.000000  0.000000
@@@descriptor:
                        4:[(Py_blue)]
       coefficients_001:     -0.9335800000E+00
          Intercept_001:     -0.1776356839E-14
         RMSE,MaxAE_001:      0.3183483133E-14    0.7105427358E-14
================================================================================
Wall-clock time (second) for this DI:          0.08
DI done!
```

Figure 50 – SISSO output file (minus initial condition)

Notice that the coefficient can be rewritten as the ratio of velocities,

$$-0.93358 = {}^{V_{y,red}}\!\!\big/\!_{V_{y,blue}} \tag{20}$$

Therefore, SISSO has obtained Eq. (10), which has the same fitting error as Eq. (12), and simplifies to Eq. (12) when the initial conditions are reintroduced. SISSO results can be used to obtain a good initial guess at the form of a function, and although the form of the function is not unique, there are multiple forms of the same function that have the same computational complexity (e.g. multiple ways to write an expression of associative terms). If needed, a user may find it possible to further simplify the equation produced by SISSO by hand, or with their own computer program.[291]

Going back to analytical derivations, consider the cases of properties that remain constant when the flock is stable and not interacting. In this case, suppose a boid belonging to the red flock is denoted with subscript *i*, while a boid belonging to the blue flock is denoted with subscript *j*. Since all boids have the same base speed, the velocity equations can be written as a set of three vector equations (the relationship between the two boid-level variables is implied):

$$\begin{cases} \vec{V}_{red,i} = \vec{V}_{red,F} \\ \vec{V}_{blue,j} = \vec{V}_{blue,F} \\ \vec{V}_{red,F} = \mathrm{R}\vec{V}_{blue,F} \end{cases} \tag{21}$$

where the *F* subscript is added to distinguish the flock's velocity from the velocity of its boids, V is the velocity, the arrow over the V indicates that it is a vector, and R is a constant

[291] Time does not permit developing a rigorous algorithm for that purpose (some obvious simplifications, such as terms that reappear in the equation, have been taken into consideration).

2x2 rotation matrix. As before, Hypothesis 2 is supported (the equations have the same RMSE and C_T, and so the numerical criteria do not find an emergent behavior). However, these equations bring up a broader issue that is not explicitly dealt with in the numerical criteria. The flock velocity is a linearly computed property as shown in the following equation,

$$\vec{V}_{red,F} = \frac{1}{n}\sum_{i=1}^{n}\vec{V}_{red,i} \tag{22}$$

In terms of the manner in which $\vec{V}_{red,F}$ is computed, the equation for $\vec{V}_{red,F}$ is never identical to the properties of the boids (it is a multivariable equation that cannot be simplified into an exact copy of the corresponding boid-level equation). However, when the flock is stable and non-interacting, its numerical values are identical to the speed and heading of the boids. Therefore, on quantitative grounds, one could write,

$$\exists\ t^*\ s.t. \forall\ i \in [1,n]\ \begin{cases}\vec{V}_{red,F} = \vec{V}_{red,i} & t \geq t^* \\ \vec{V}_{red,F} \neq \vec{V}_{red,i} & t < t^*\end{cases} \tag{23}$$

where *n* is the number of boids in the red flock. In other words, there exists a time interval (any interval of stable flight beginning at *t**) where the heading of the flock is equal to the heading of any member boid (thereby also being directly proportional to the heading of any boid in the opposing flock), and there exists another time interval in which the headings are strictly not equal to each other. Therefore, for some time intervals, linearly computed properties can be quantitatively indistinguishable from boid-level properties. This thesis will take the position that, since the equations are known to be different *a priori*, the velocity of the flock is not equivalent to the velocity of the boid (therefore, velocity satisfies both distinctiveness criteria). It is left to the user of these numerical criteria to beware of this potential ambiguity when analyzing quantitative data. However, as with position, since the RMSE and C_T of the various equations are the same, the numerical criteria find that this is not an emergent behavior during this time interval.

Not all properties have this ambiguity. Slope is a property of the flock (i.e. the self-organized object). A slope is mathematically undefined for a boid (a single point), and so it passes the distinctiveness test, but fails the interaction test (RMSE, C_T) and cannot be considered an emergent behavior during this time interval. This means that although the slope is a valid geometric property, it does not appear to serve a purpose, and so, is not emergent. Strictly speaking, boids do not have a length, but it is not difficult to imagine that since a one-dimensional point can be embedded in a two-dimensional space then a boid has a length equal to zero. Stretching the imagination in this way, however, will not be explored in this thesis. What is more important is that length, like slope, is constant during stable flight, and so the equations relating it to the length of the opposing flock, or perhaps the speed of the opposing boids[292] will have the same RMSE and C_T. Therefore, these properties do not satisfy the interaction criteria, and are not considered emergent behavior during this time interval.

To be clear, saying that these properties "are not considered emergent behavior during this time interval" should be interpreted to mean that the numerical criteria do not

[292] Since the approach in this thesis does not constrain the coefficients of the equations in any particular way, it is possible to write an equation where the length of the red flock is directly proportional to the speed of the opposing boid (since both are constant). The only difference between this and the relationship between the lengths of the two flocks is the units of the leading coefficient (and the amount of credulity).

detect an emergent behavior. Whether or not a property is an emergent behavior[293] is not solely a consequence of the time-interval during which it was detected.

6.2.2 Interaction Phase Data

The interaction phase begins once a boid from one flock can see any boid from the other flock. The behavior rules will cause this boid to begin to adjust its position, and as more boids enter into view, each flock will destabilize further and their respective shapes changes over time. Although cases where the flocks separate (even for a single iteration) are omitted, some perturbations may be so severe that the flock can no longer be referred to as a straight line. A flock is classified as line because of its initial and final shape over the course of the simulation, not this time interval. Therefore, this interval is best understood as the interval in which a flock experiences a destabilizing perturbation, and ending with the flock in a destabilized form. The numerical criteria were not intended to focus solely on this time interval because a change in property is only meaningful after the flock has resumed its original shape (within the confines of the ontology in this thesis)[294] but it is informative to see how they respond to this subset of the data. For the purposes of generating equations, only boids that actually participate in the interaction are considered for FvB equations.

Of the six properties considered here, only speed (the norm of the flock velocity) has Pareto Fronts that dominate their FvB counterparts on the training data set and the

[293] The terms property and behavior are used interchangeably for simplicity. Any time-varying property can be an emergent behavior. See Section 2.3. This thesis does not examine the case of constant properties.

[294] Yet another simple (albeit morbid) analogy: the logic is that if two people eat food, but one of those two eats food that is poisonous, one must wait until after the food works through their bodies to determine whether it is appropriate to continue referring to each person as a person rather than a corpse. The end-state of the system must possess some crucial quality to prove that it persists. Here that quality is a linear arrangement.

testing data set (see Figure 54). In a careless application of the numerical criteria, this would be taken to mean that an emergent behavior has been found (a false positive). Nevertheless, this would appear to be reassuring because speed can only be affected by interactions, and since multiple boids are simultaneously involved, one would expect it to become impossible to accurately model the change in flock speed as a function of any single boid's behavior from the opposing flock. According to this line of reasoning, the results from the numerical criteria present a false-positive suggesting that the criteria, as presented in CHAPTER 4, are not sufficient (more conditions are needed to avoid false positives). In addition to this, one of the models in the FvF Pareto Front of Figure 54b presents a peculiar case not previously considered:

$$\text{FvF:}\quad L_{F1,f} - L_{F1,0} = -0.06 + 0.005\left(M_{F1,0} + M_{F2,0} + \left|M_{F1,0} - M_{F2,0}\right|\right) \qquad (24)$$

This can be rewritten as an implicit function with quadratic length terms, or, more interestingly, as a piecewise continuous function,

$$\text{FvF:}\quad L_{F1,f} - L_{F1,0} = \begin{cases} -0.06 + 0.01M_{F1,0} & M_{F1,0} \geq M_{F2,0} \\ -0.06 + 0.01M_{F2,0} & M_{F1,0} < M_{F2,0} \end{cases} \qquad (25)$$

What Eq. (25) reveals, which Eq. (24) quite effectively masks, is that it is possible to obtain a model that satisfies the definition of an interaction equation for one part of its domain, while appearing to violate that definition in another part. The portion of the equation given by $-0.06 + 0.01M_{F1,0}$ would completely violate the definition of an interaction equation were it not for the condition that $M_{F1,0} \geq M_{F2,0}$. Setting aside the question of whether it makes sense that the values of the initial slopes of the two lines should fully explain the subsequent change in their lengths,[295] this possibility further adds to the difficulties of finding an emergent behavior model. Since the piecewise defined function cannot eliminate $M_{F2,0}$ completely from the equation, it pushes the definition of an interaction equation to a very interesting extreme. It remains to be determined in some future study, whether the definition of an interaction provided in this thesis is robust enough to be used as it has. Nevertheless, since this special case does not completely eliminate the variable due to the other flock, $M_{F2,0}$, from the equation, there is no need to discard the current definition of an interaction equation just yet.

Heading, like speed, only varies during interactions. However, heading displays the exact opposite result (Figure 55). Although it is not beyond reason to suppose that a single boid can heavily influence the heading of the flock, it seems odd that models built with the properties of a single, *randomly-selected* boid have more explanatory power than models built from flock-level variables. Perhaps the variability in heading among boids in the opposing flock is too small to make a difference in the error of the model. Nevertheless, this piece of information should be kept in mind as the analysis progresses. Consider also the equations for the single-point Pareto Fronts in Figure 55b.

$$\text{FvF:} \quad H_{F1,f} - H_{F1,0} = -0.135 + 0.040\left|n_{F2}H_{F1,0} - M_{F1,0} + H_{F2,0}\right| \tag{26}$$

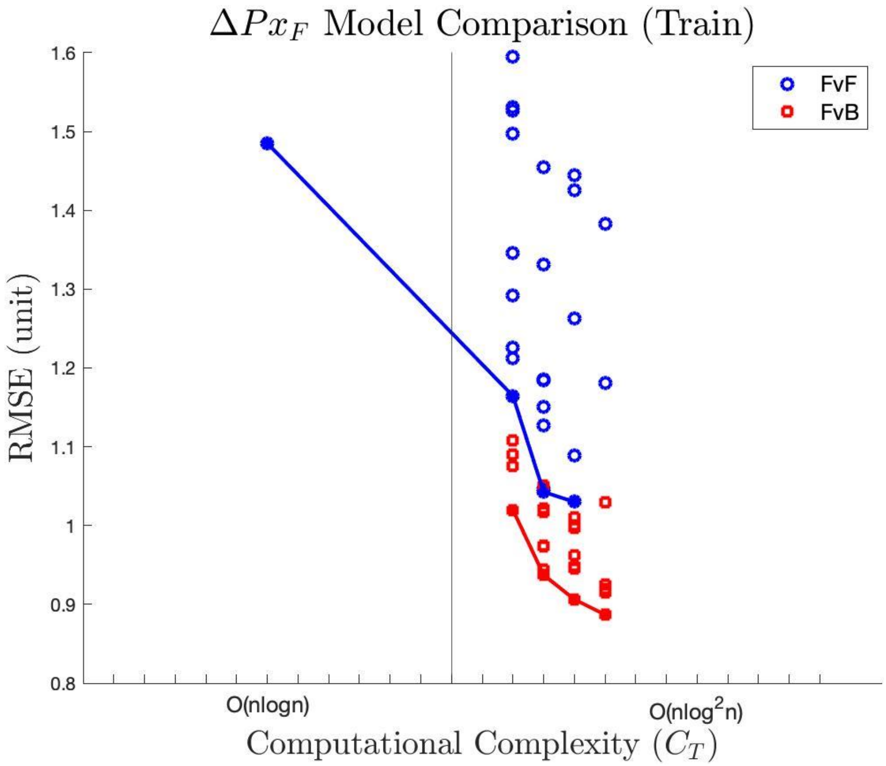
$$\text{FvB:} \quad H_{F1,f} - H_{F1,0} = 0.141 - 0.032\left|n_{F1}H_{2,0} - M_{F1,0} + H_{F1,0}\right| \tag{27}$$

[295] In practice, such an equation would have to be tested empirically, or, due the scope of this thesis, more simulations. It is not essential for the purposes of testing either hypothesis because the underlying argument is that the proper selection of variables will have a bigger impact on RMSE and C_T than the form of the equation (given that SISSO can obtain a wide range of functional forms).

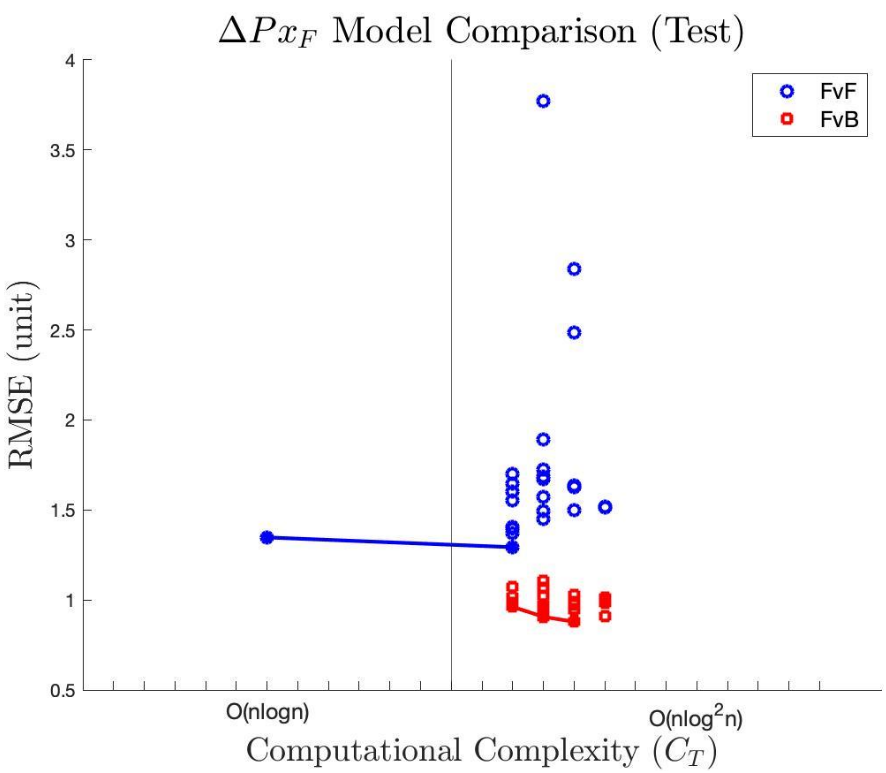
Note that the variable n is the number of boids in the flock. These equations are striking in their similarity and apparent symmetry (from the sign flip, to the arrangement of terms). Even more remarkable is that a similar result appears in a different time interval (see Section 6.2.5 and 6.2.6 for more discussion).

Besides the heading and speed, the position variables, which are the integrals of the velocity vector components, do not consistently show either result (the x-position shows weak non-dominance, while in the y-position the FvB models strongly dominates the FvF models). This thesis does not take a stance on whether the integral of an emergent behavior is also an emergent behavior (or its derivative, for that matter). This is an important question to answer once a criteria for emergent behavior detection have been established. For now, it suffices to test whether these findings contradict the definitions provided here. This observation does not present any obvious contradiction to the criteria. Another important factor to consider is whether the criteria are adequate for testing vector-valued properties. Since the two position components falsify the criteria in different ways (as opposed to consistently) then it is unclear how the vector-valued property should be labelled (weakly non-dominating versus dominated). Furthermore, since speed and heading are scalars, it is not obvious that converting them back to velocity vector components would provide useful information. Consider also that speed and heading are two scalars, and so can be formulated as a vector (or at least as a point in two-dimensional space). Would the combination of a property that satisfies the criteria and a property that fails the criteria produce a vector that satisfies the criteria? Is the fact that speed satisfies the criteria enough to argue that velocity is an emergent behavior? These challenges do not falsify any hypothesis in general, but certainly shows that there may be a limitation when it comes to vector-valued properties.

The FvF models for slope dominate the FvF models over the training data, but not the test data. Although all the models experience a significant increase in error on the test data, it seems that one FvB model remains accurate enough to disqualify this property from being considered an emergent behavior within the context of this time interval. The behavior of the models for flock length are a bit more nuanced, but the ultimate conclusion is the same. Since the time interval is inappropriate for emergent behavior testing, the only significance of this finding is that the geometric properties one might expect to stand out do not necessarily do so. Thus far, the numerical criteria do not have an obvious bias toward any particular property. Naturally, this raises the question of whether the numerical criteria are capable of finding anything at all, but that question cannot be addressed with this data alone.



(a)



(b)

Figure 51 – Pareto optimal flock x-displacement models due to interaction (a) training data (b) test data

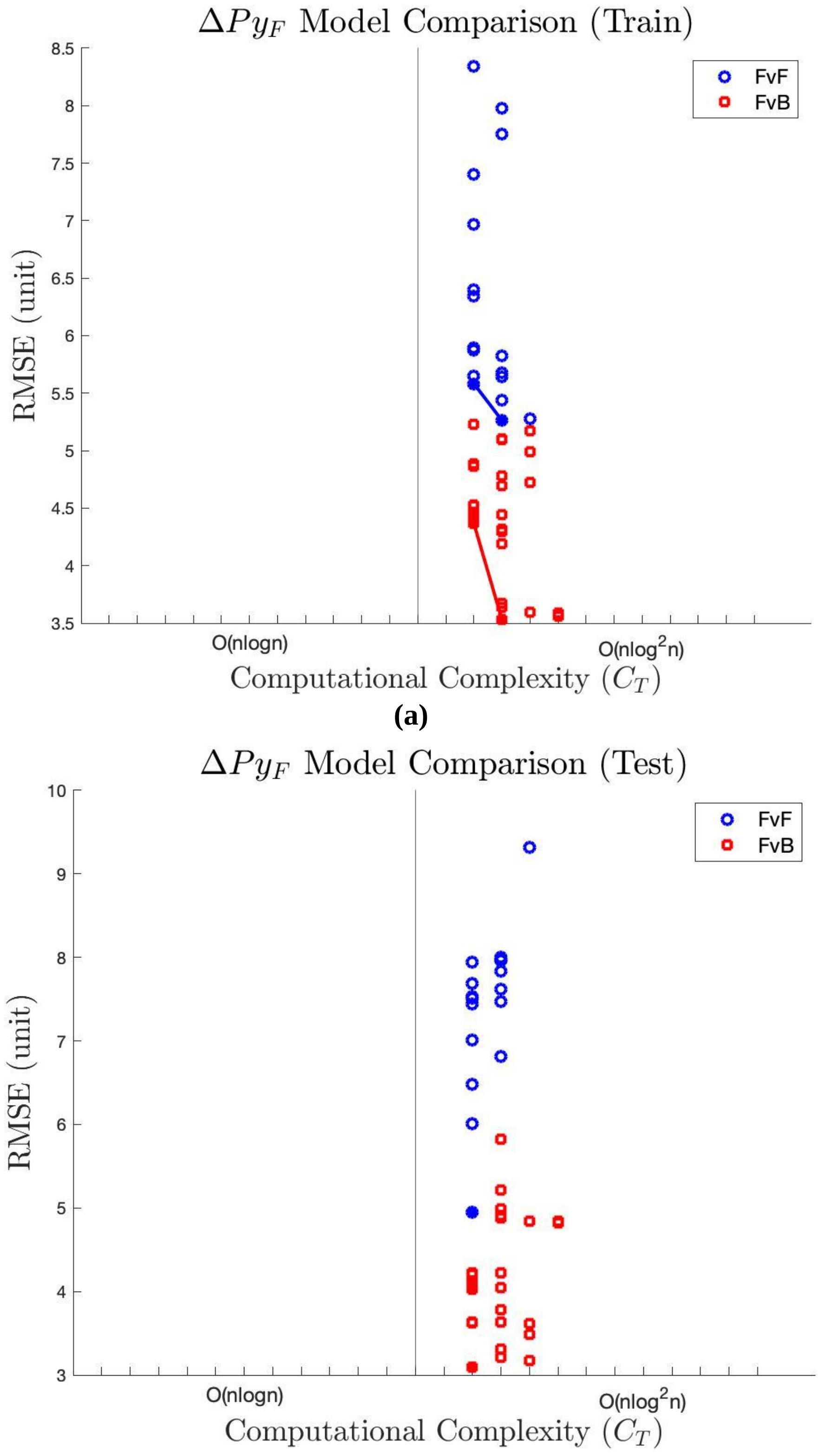
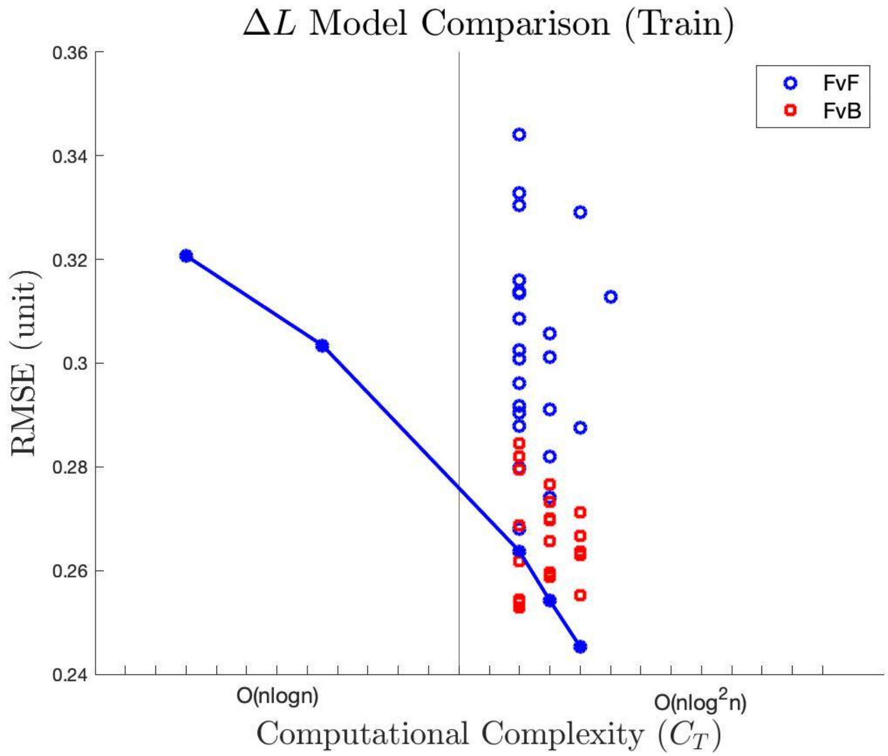
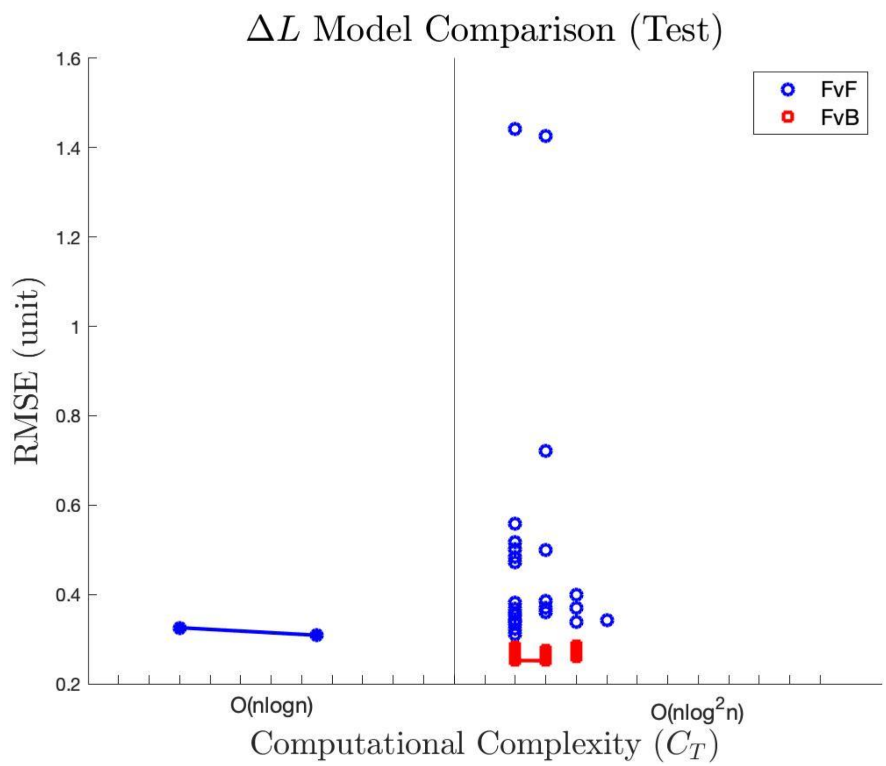


Figure 52 – Pareto optimal flock y-displacement models due to interaction (a) training data (b) test data

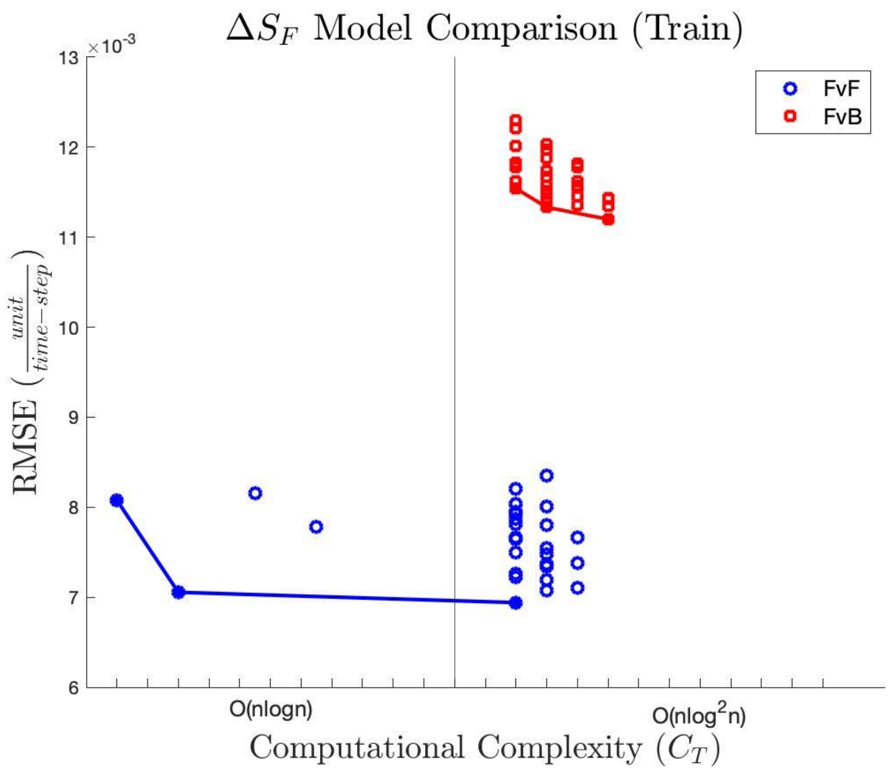


(a)

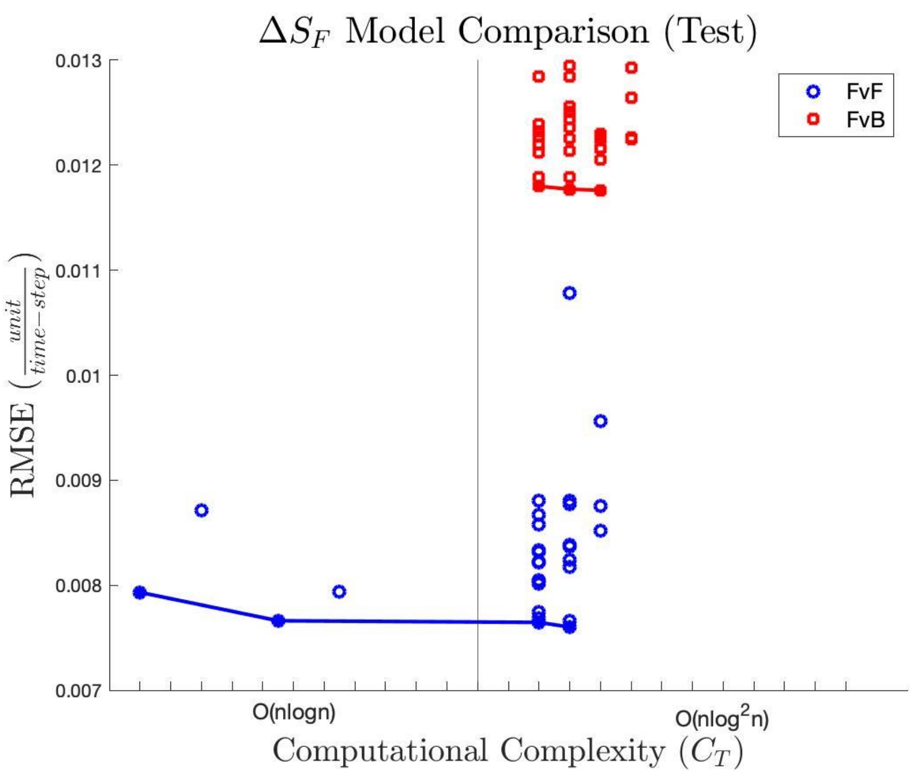


(b)

Figure 53 – Pareto optimal flock length models due to interaction (a) training data (b) test data

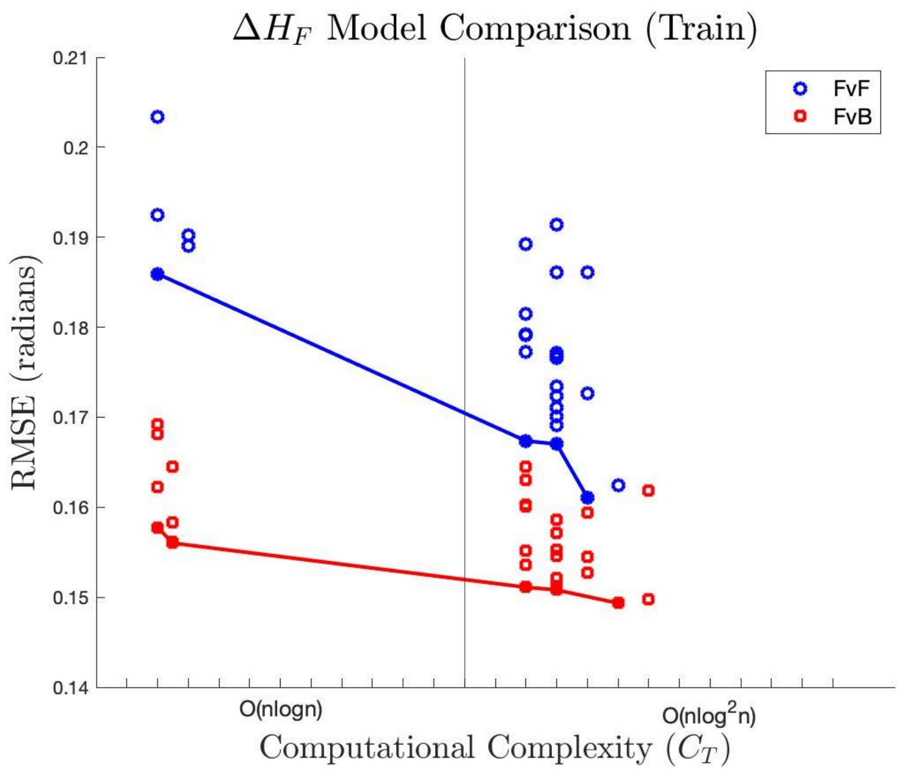


(a)

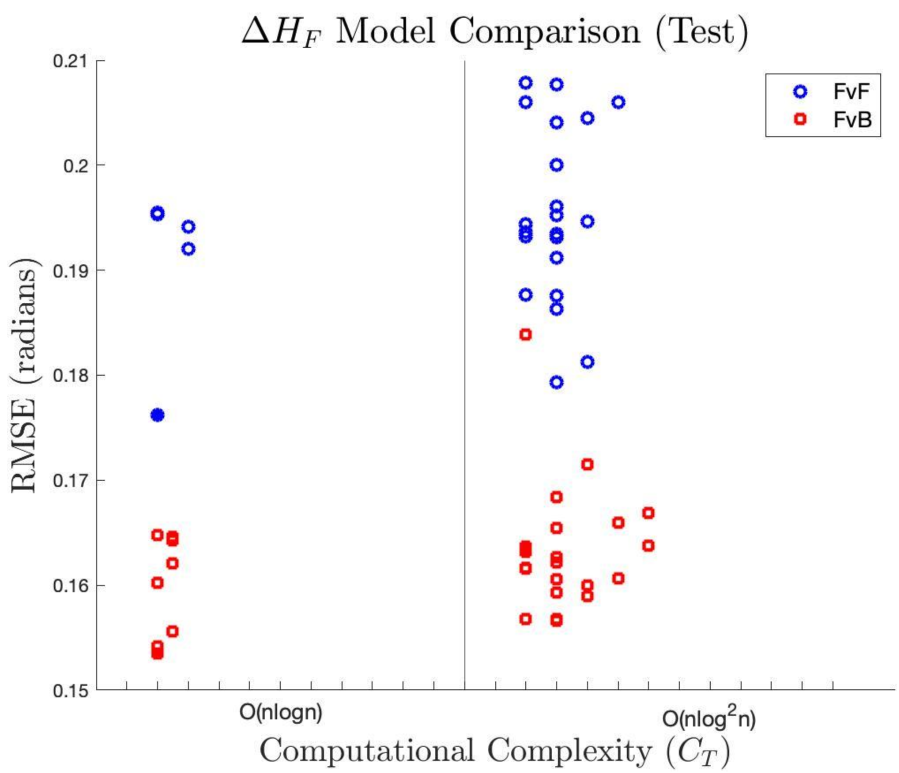


(b)

Figure 54 – Pareto optimal flock speed models due to interaction (a) training data (b) test data

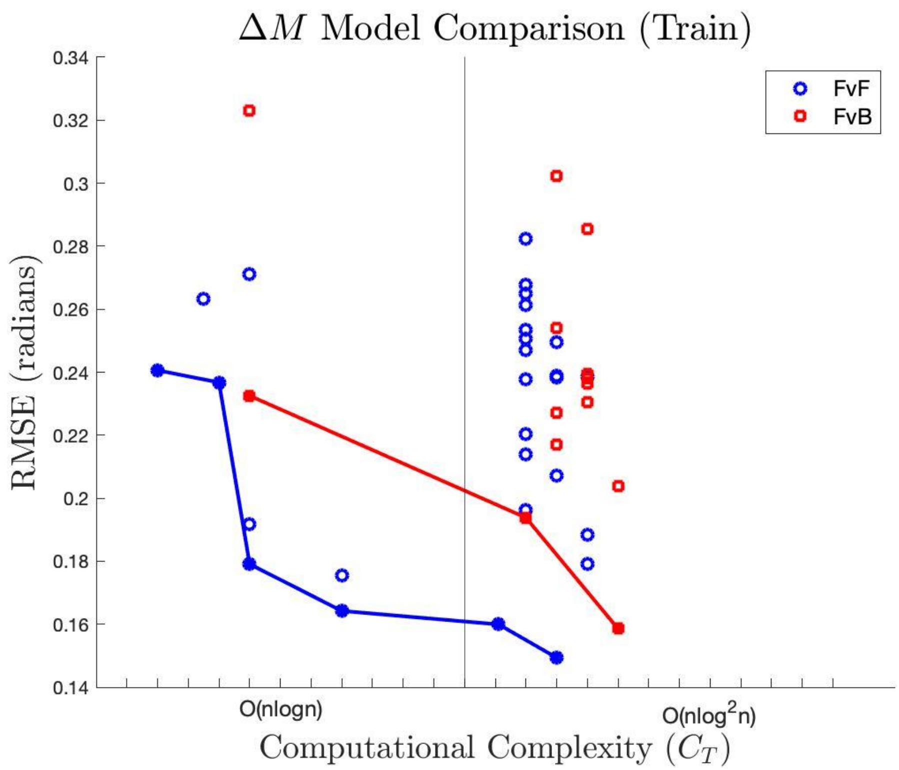


(a)

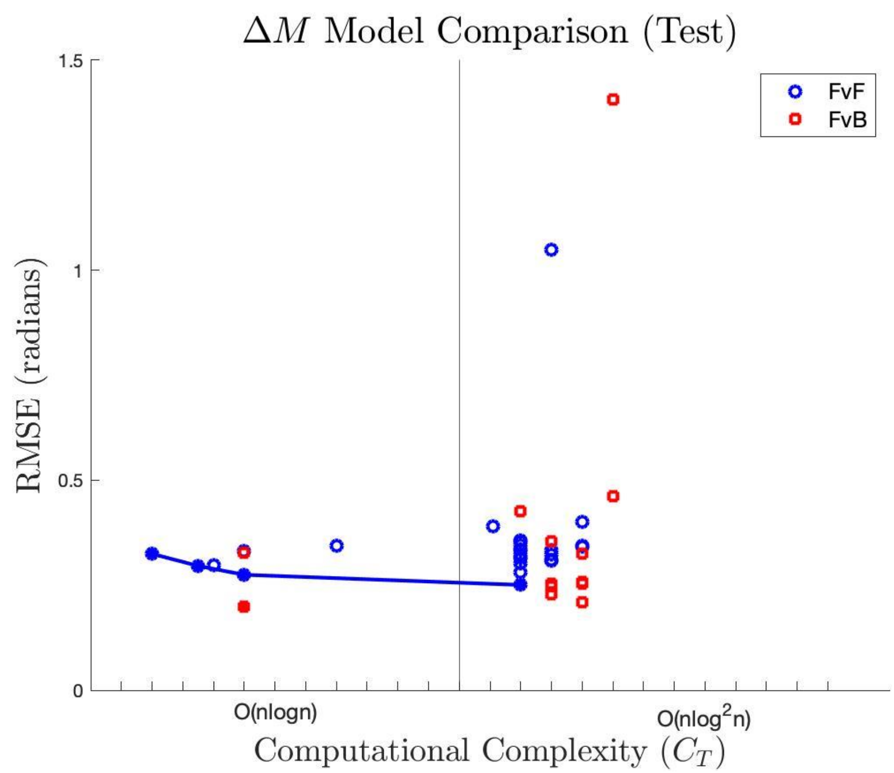


(b)

Figure 55 – Pareto optimal flock heading models due to interaction (a) training data (b) test data



(a)



(b)

Figure 56 – Pareto optimal flock slope models due to interaction (a) training data (b) test data

6.2.3 Re-stabilization

As with Section 6.2.1, if the numerical criteria suggest that an interaction has occurred, then Hypothesis 2 is falsified because there is no interaction between flocks during this phase. In other words, the question answered here is whether spurious[296] regressions satisfy the numerical criteria. For all but one of the properties, the FvF models dominate the FvB models, therefore, the hypothesis is in fact falsified. This means that criteria must be added that specify the time interval over which the sampling must occur. Although these were not explicitly stated in the criteria, this was expected. Therefore, the recommendation to add such criteria will be made later in this document, and this analysis will proceed in search of more interesting ways to falsify Hypothesis 2.

Unlike Section 6.2.2, the FvF models of the x-coordinate dominate the FvB models. The discrepancy between the result for the x-coordinate and the y-coordinate remains. However, this time the y-coordinate FvB model causing this result exhibits the same odd behavior seen in Eq. (24).

FvB: $$Py_{F1,f} - Py_{F1,0} = -50.36 + 10.1\left(H_{2,0} + \left|H_{2,0} - M_{F1,0}\right|\right) \tag{28}$$

FvB: $$Py_{F1,f} - Py_{F1,0} = \begin{cases} -50.4 + 10.1 M_{F1,0} & M_{F1,0} \geq H_{2,0} \\ -50.4 + 10.1\left(2H_{2,0} + M_{F2,0}\right) & M_{F1,0} < H_{2,0} \end{cases} \tag{29}$$

When the H2,0 variable is eliminated from the equation, the FvB equation no longer qualifies as an interaction. Note, also, that the constant in this equation is -50.4, which is

[296] Spurious in the sense that it is not an emergent behavior. The regression found by SISSO may, in fact, be predictive (i.e. not spurious in the usual statistical sense).

approximately half of the entire spatial domain. Since the domain of the angular variables is $[0,2\pi)$, this equation covers the entire y-domain of (-50.5, 50.5).[297] The RMSE in the y-coordinate (see Figure 58) raises the question of whether the Pareto Front is the correct approach to take, not because the magnitude of the error is high (although that is problematic for other reasons), but because, in principle, the Pareto Front can include models whose errors are arbitrarily high. SISSO performs the task of filtering out absurd models (based on the training data set), so this extreme cannot be reached given the tools used in this thesis. Therefore, the symbolic regression tool is at least as important as the Pareto Front itself. Returning to the errors, note that there are very few points in the x-coordinate plots (Figure 57). Most of the results produced on this training data set were discarded because their maximum absolute error on the test data set was unacceptable (some errors in position exceeded the size of the domain itself). This additional precaution is taken in every data set presented in this chapter. Therefore, the Pareto Front can be used as a mechanism for studying families of models, so long as additional steps are taken to filter invalid candidates.

In this time interval, the models for length exhibit a new noteworthy feature. For the length models, the FvB Pareto Front appears to be weakly non-dominating with the FvF Pareto Front due to its leftmost point (Figure 59). The relevant equations produced by SISSO (appear in the O(nlogn) category models in Figure 59b) are:

[297] Values exceeding either limit are understood to be "wrapping around" the domain. The data was manually "unwrapped," and so "re-wrapping" must be done manually. On second thought, a re-parametrization of the spatial domain would have probably resulted in much friendlier equations. Also, a deep dive into the Netlogo documentation is required to determine whether the upper or lower bound is closed. That detail was not a problem in this analysis because the boids only cross the boundary once (imposed by the time-interval/speed).

FvB: $$L_{F1,f} - L_{F1,0} = 0.291 - 1{,}682\left(S_{F1,0}\right)^6\left|H_{2,0} - H_{F1,0}\right| \tag{30}$$

FvF: $$L_{F1,f} - L_{F1,0} = 0.866 - 14{,}938\left(S_{F1,0}\right)^6\left({S_{F2,0}}/{S_{F1,0}}\right)$$

(*sim*) $$L_{F1,f} - L_{F1,0} = 0.866 - 14{,}938\left(S_{F1,0}\right)^5 S_{F2,0} \tag{31}$$

In the case of Eq. (31), SISSO did not fully simplify the equation, which introduced an extra division operation into the equation. The manually simplified result is indicated as (*sim*). After simplification, it becomes clear that the FvF models dominate the FvB models. Although both models have a nonlinear term with a large exponent, which is typically a cause for concern in engineering modeling scenarios, the massive coefficient in Eq. (31) is particularly alarming (again, based on experience).[298] Since this is only one of a full set of equations, and the claim of emergent behavior is based on the efficacy of the full set, this does not falsify any hypothesis. Furthermore, any model of a physical system needs to be validated, so the fact that its coefficients vary wildly in their magnitude is not, in and of itself, the primary driver for any particular action. Nevertheless, an extension of this method to a real-world test case would probably do well to include additional reports on the range of magnitudes of the coefficients in the equations, as such reports would help subject matter experts quickly identify problematic models.

Regarding slope (Figure 62), one of the models that is contained in both the training set and testing set Pareto Fronts contains a problematic feature that can be used to disqualify it as a viable model without generating new simulation or experimental data.

[298] The large coefficient in Eq. (30) is not much better.

FvF: $$M_{F1,f} - M_{F1,0} = 1.51 - 0.6658\left(M_{F1,0} S_{F1,0}\right)\sqrt{M_{F2,0}} \tag{32}$$

The model in Eq. has RMSE = 0.462 on the training set, and RMSE = 0.42 on the test set. The square root term in Eq. (32) indicates that whenever the initial slope of the opposing flock is zero (i.e. a vertical line), the change in the flock's slope will always be 1.51 radians. There is no reason to believe such a result, because there is nothing special about a vertical flock. If this point were to be removed from the data set (as it would in practice), then the FvB and FvF Pareto Fronts are weakly non-dominating. This does not change the conclusions drawn from this section because the other data in this section already falsified Hypothesis 2.[299] What this does show, however, is that the models generated by a symbolic regression tool must be approached with skepticism. Although these tools are useful and powerful, they are capable of finding models that are nonsensical even on a purely abstract problem such as this. Therefore, the user attempting to implement the method in this thesis (or some variant of it) must include an additional mechanism for removing such models from the set of candidate models.[300]

For this time interval, the speed plots (Figure 60) greatly resemble those of the interaction phase (Figure 56). Given that no interaction is occurring here (between flocks), it is clear that some criteria needs to be added that guarantees the time interval under consideration is appropriate (otherwise, many irrelevant nonlinear behaviors will appear to

[299] It will be retained in the results for the sake of consistency.
[300] Based on this observation, one suggestion would be to iteratively substitute the minimum and maximum value of each variable into the equation to see what happens. Zeros are a great way to find breaking points.

be emergent behaviors). Note that there are only three models for speed that fall into the O(nlogn) complexity bin.

FvF: $$S_{F1,f} - S_{F1,0} = 0.043 - 26.66\left(S_{F1,0}\right)^4 \tag{33}$$

FvF: $$S_{F1,f} - S_{F1,0} = 0.074 - 4.13\left(S_{F1,0}\right)^2\sqrt{S_{F1,0}} \tag{34}$$

$$S_{F1,f} - S_{F1,0} = 0.073 - 4.15\left(S_{F1,0}\right)^2\sqrt{S_{F1,0}} \ldots$$

FvF: $$\ldots + 1.96 \times 10^{-4}\frac{H_{F2,0}}{S_{F2,0}\left(H_{F2,0} - M_{F1,0}\right)} \ldots \tag{35}$$

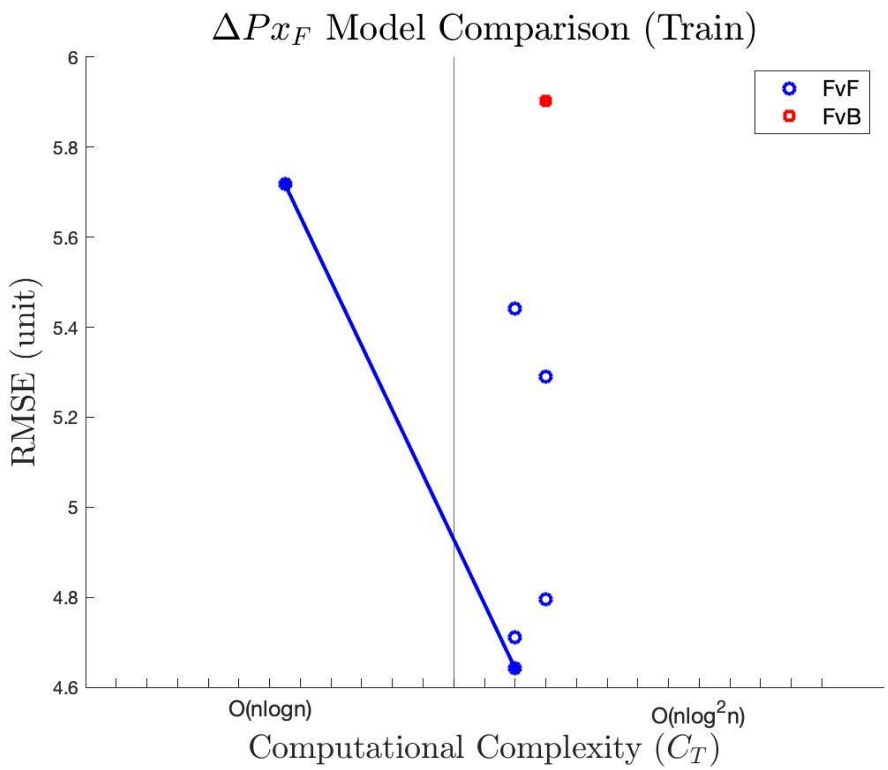
$$\ldots + 4.67 \times 10^{-4}\frac{M_{F1,0} + M_{F2,0}}{\left|M_{F1,0} - H_{F2,0}\right|}$$

Equations (33) and (34) are Pareto Optimal models. However, neither model qualifies as an interaction equation, and so cannot function as an emergent behavior equation. Once again, the results of the symbolic regression tool cannot be accepted blindly.

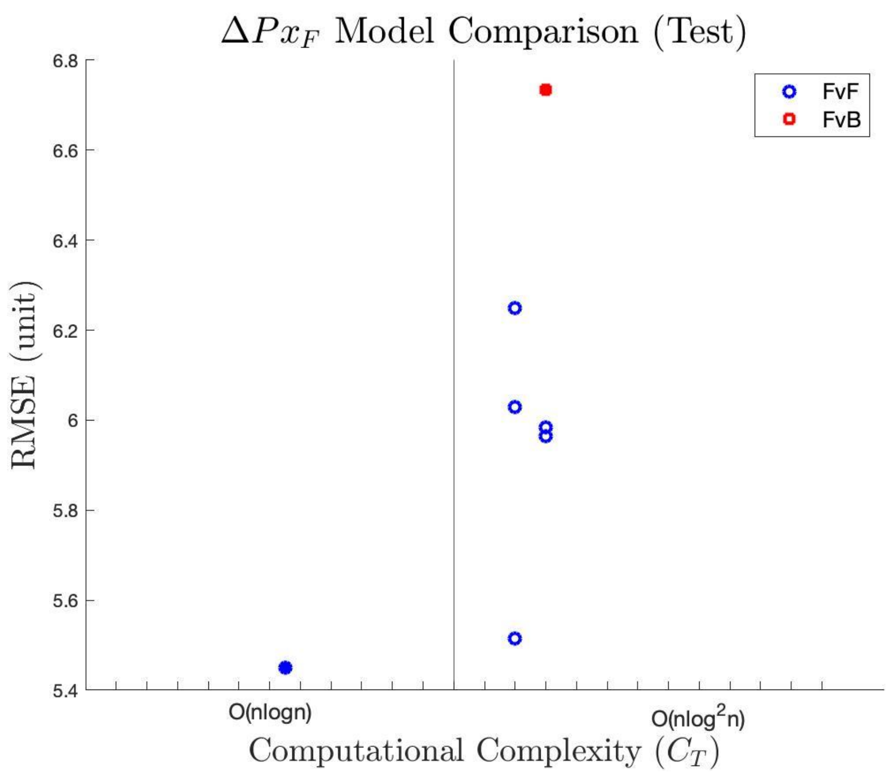
Due to the manner in which the FvF heading models dominate the FvB models, and the preceding discussion, there is no need to discuss the data beyond the single FvF Pareto Optimal point in Figure 61b.

FvF: $$H_{F1,f} - H_{F1,0} = 0.104 - 0.067\frac{L_{F2,0}}{L_{F1,0}\left(H_{F1,0} - H_{F2,0}\right)} \tag{36}$$

Eq. (36) presents the very interesting case of an equation that diverges. If two flocks were initialized such that one flock was directly behind the other, given the same heading, and the lead flock was visible to the aft flock, the change in heading is mathematically undefined according to this equation. There are at least two possible ways to arrive at this result in an ordinary simulation. First, there is the “miraculous” scenario. The probability of obtaining exactly two linear flocks with the same heading experiencing an interaction from a random initialization is zero (called miraculous because there are infinitely many more alternatives). Second, there is the spurious scenario. It is certainly possible that a randomly initialized scenario, with a sufficient number of boids, will have so many boids in one place that one could draw lines through two sets of boids such that an analyst might think this equation applies. In other words, it might seem as though there are two linear flocks, one right behind the other seeming to interact, inside a larger wave of boids. If, by a statistical miracle, the first case occurred, this equation could not predict a meaningful outcome (any outcome can be justified due to division by zero). If the second scenario occurred, it would be possible to incorrectly assume the presence of interactions between lines contained inside the larger, self-organized group, and assign arbitrary outcomes to those interactions. Either case would make undermine the finding of an emergent behavior.

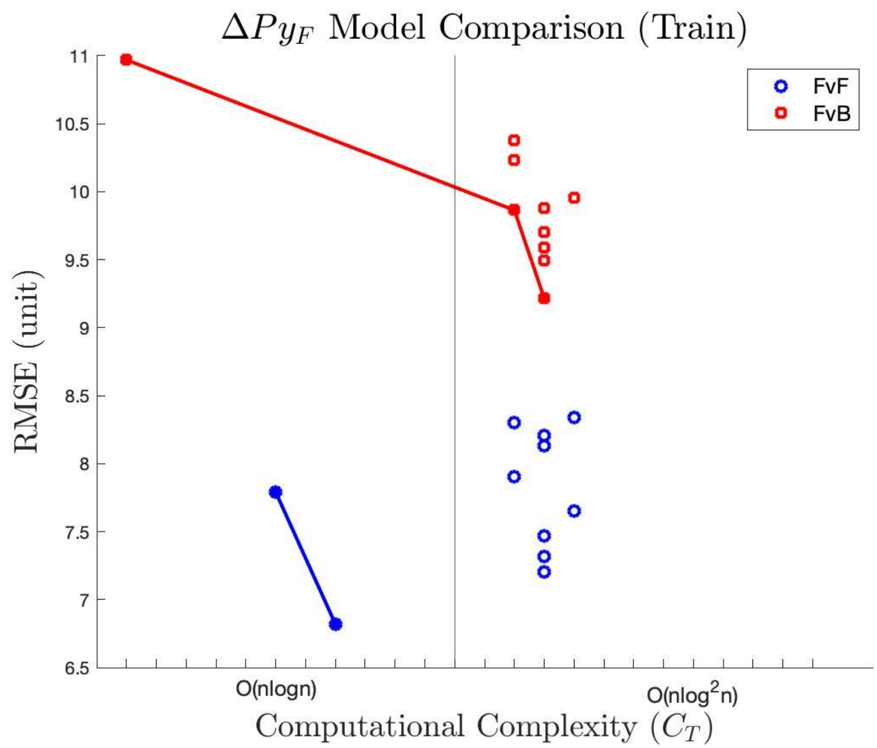


(a)

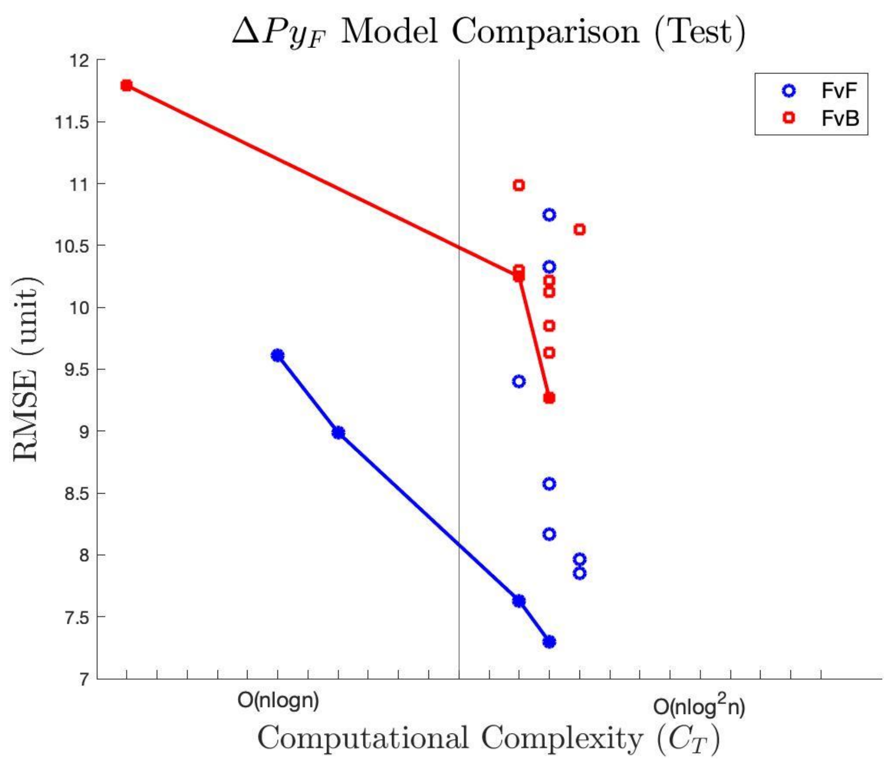


(b)

Figure 57 – Pareto optimal flock x-displacement models due to re-stabilization (a) training data (b) test data

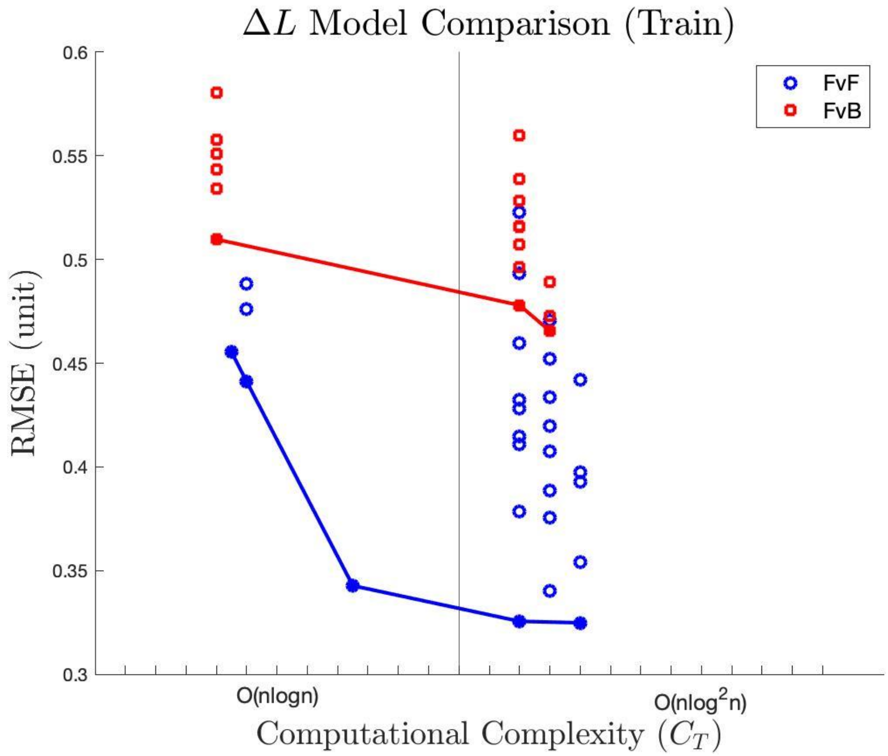


(a)

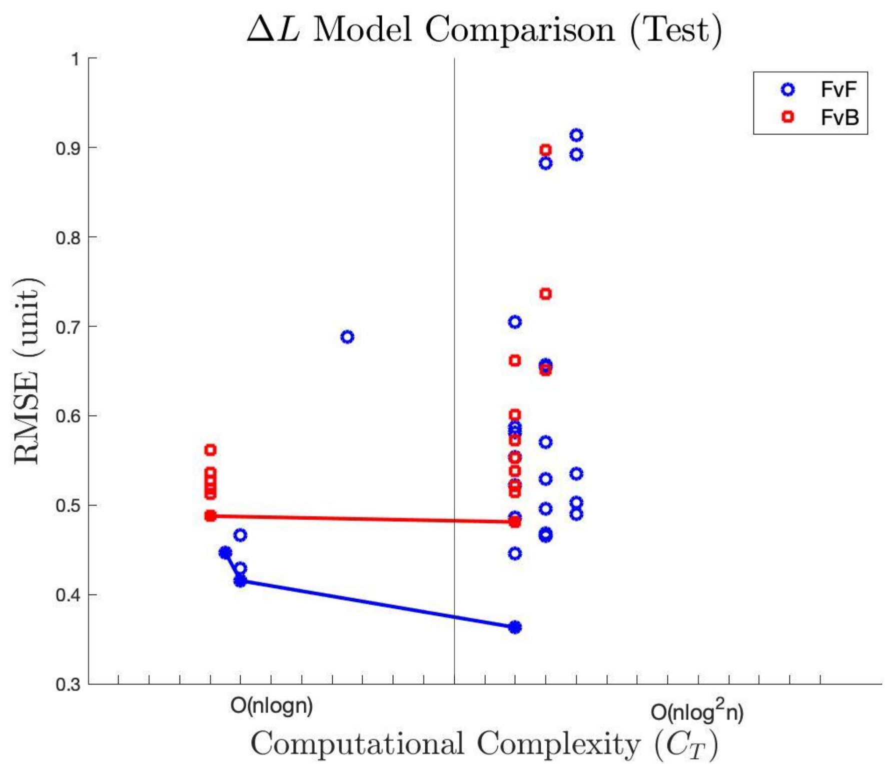


(b)

Figure 58 – Pareto optimal flock y-displacement models due to re-stabilization (a) training data (b) test data

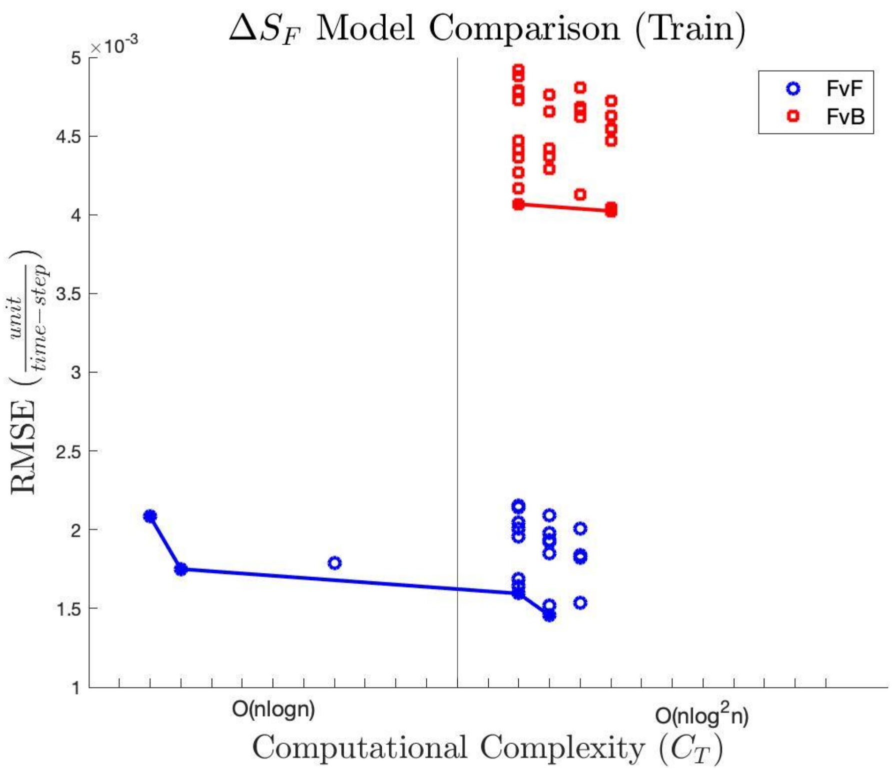


(a)

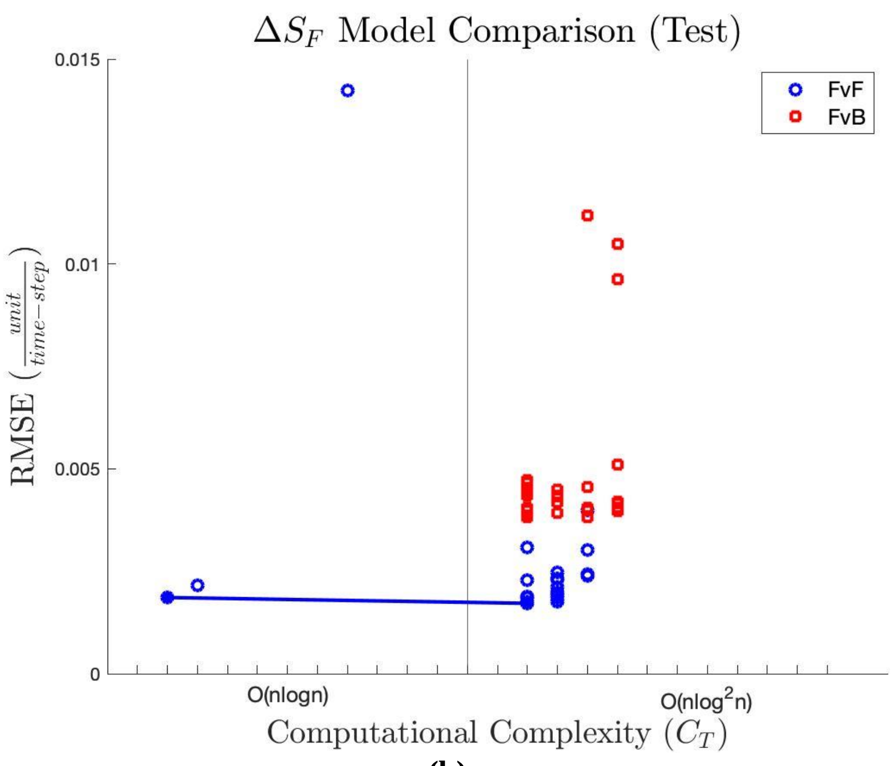


(b)

Figure 59 – Pareto optimal flock length models due to re-stabilization (a) training data (b) test data

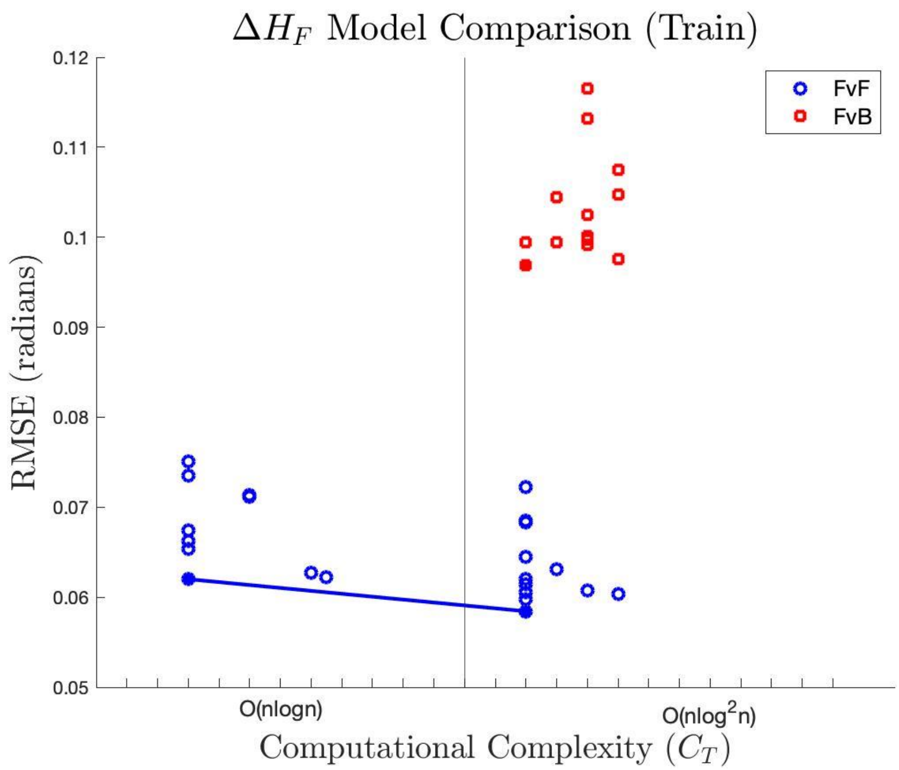


(a)

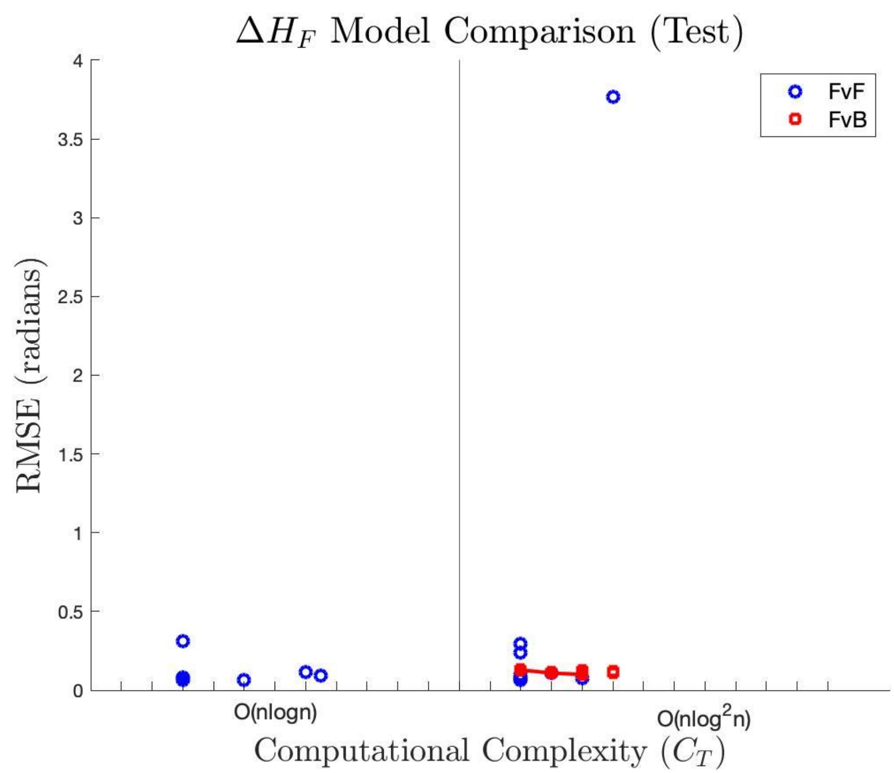


(b)

Figure 60 – Pareto optimal flock speed models due to re-stabilization (a) training data (b) test data

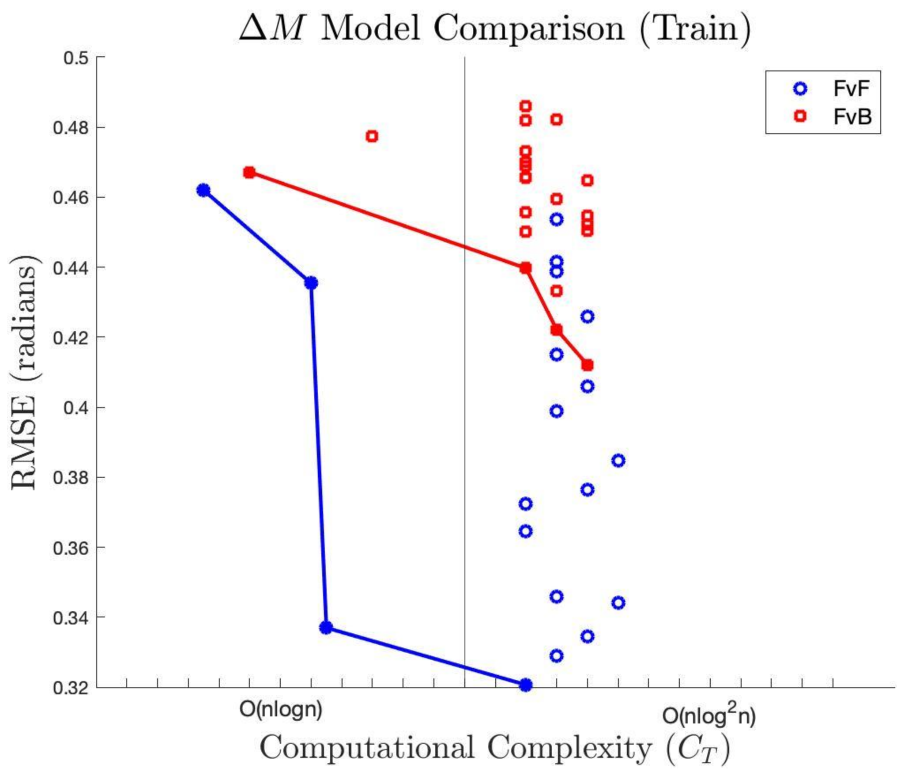


(a)

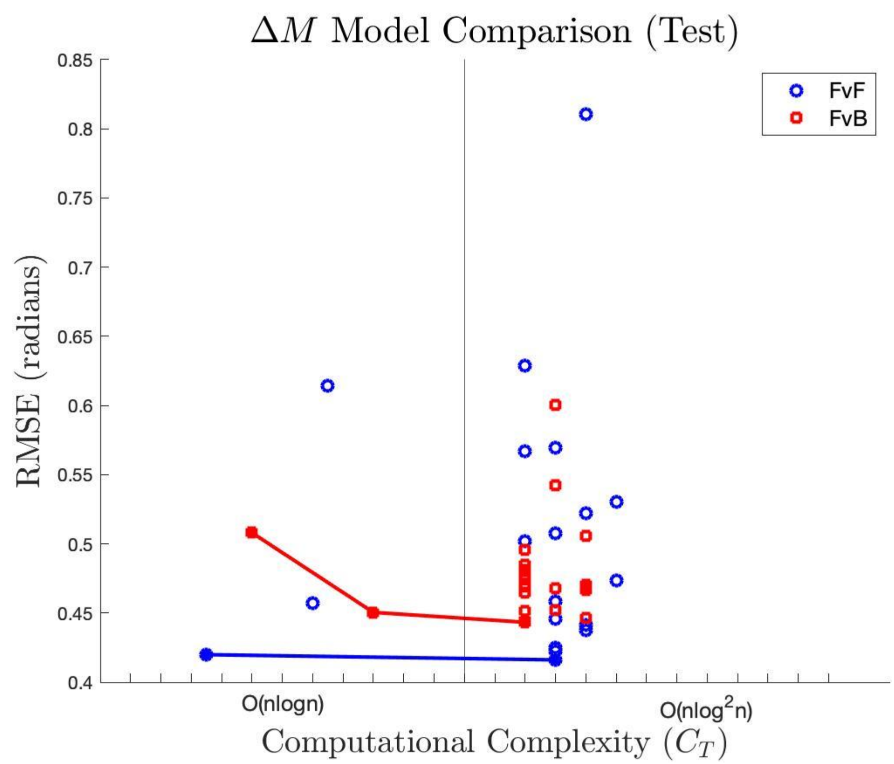


(b)

Figure 61 – Pareto optimal flock heading models due to re-stabilization (a) training data (b) test data



(a)



(b)

Figure 62 – Pareto optimal flock slope models due to re-stabilization (a) training data (b) test data

6.2.4 Interact. / Re-stab. Phase Data

Over this time interval, only the FvF models for length, speed, and slope dominate the FvB models. Since this is the time interval for which the criteria were originally intended, this means that the criteria have identified three behaviors as emergent behaviors. It is reassuring that two of the three behaviors pertain to the geometry of the flock, while the third is a property impacted directly by interactions. Although some of the models obtained for these properties (including some Pareto Optimal models) do exhibit issues seen in previous time intervals (Sections 6.2.2 - 6.2.3), only the equations for length, and slope would change the Pareto optimality determination if they were disqualified as models. The equations for slope are:

FvF: $$M_{F1,f} - M_{F1,0} = 0.09 + 0.0156 D_{F,0} H_{F2,0}\left(H_{F2,0} - M_{F2,0}\right) \tag{37}$$

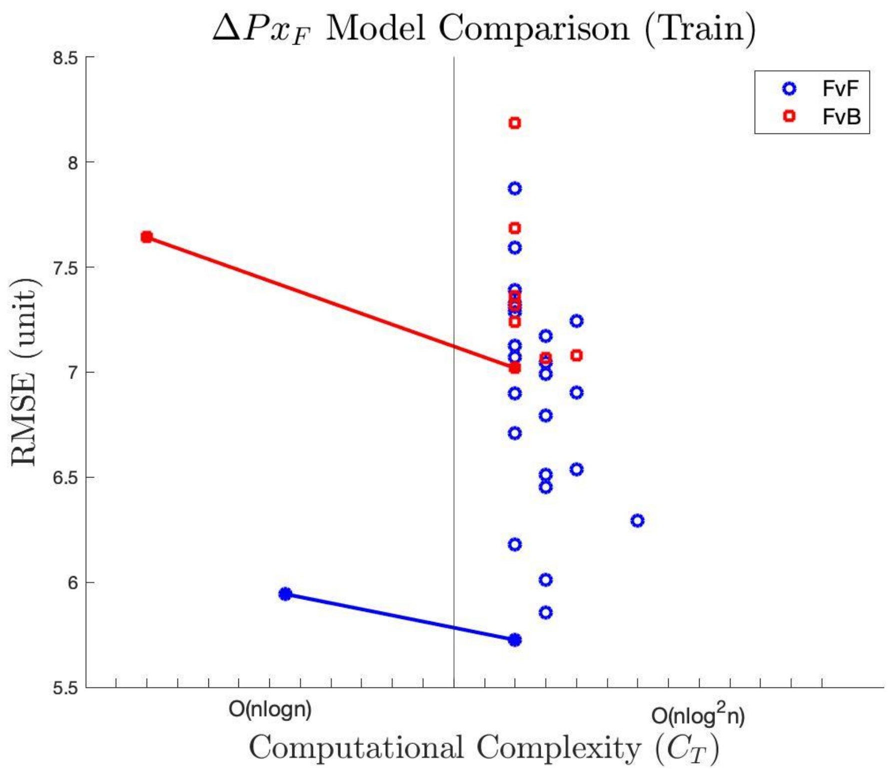
FvF: $$M_{F1,f} - M_{F1,0} = -0.214 + 132.7\frac{L_{F2,0} H_{F1,0}}{\left(M_{F1,0}\right)^6} \tag{38}$$

The variable $D_{F,0}$ in Eq. (37) represents the initial distance between flocks when the interactions begin. The equations for length are:

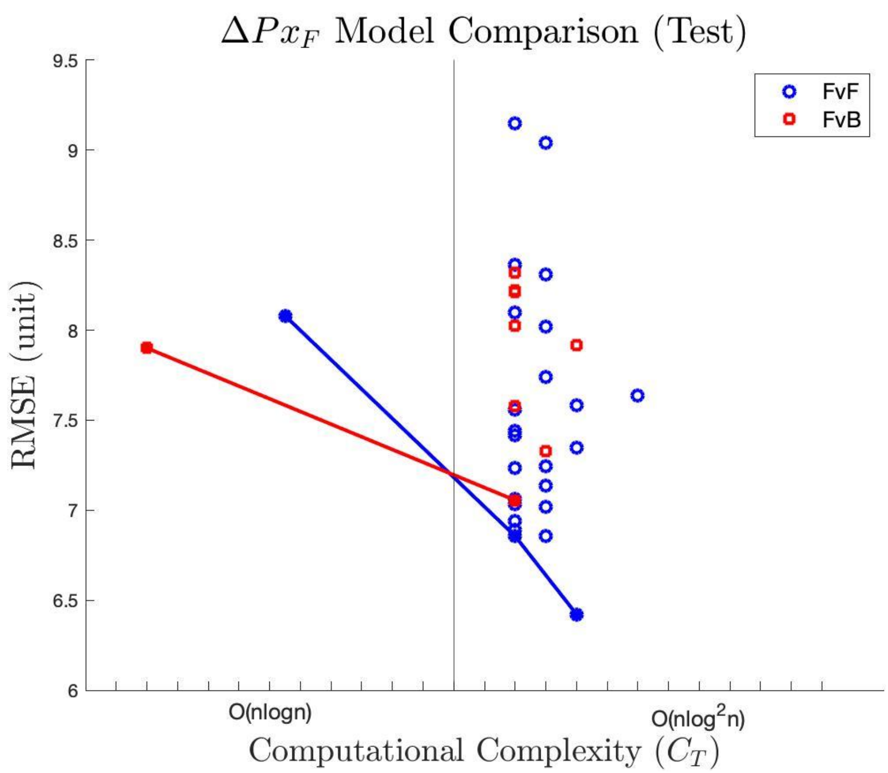
FvF: $$L_{F1,f} - L_{F1,0} = 0.163 - 0.127\frac{D_{F1,0} M_{F2,0}}{n_1 n_2} \tag{39}$$

FvF: $$L_{F1,f} - L_{F1,0} = 0.208 - 0.197\frac{L_{F1,0} M_{F2,0}}{(n_1)^2} \tag{40}$$

The variables n_1 and n_2 represent the number of boids in each flock. Whether or not these results withstand scrutiny will be discussed in Section 6.2.6.

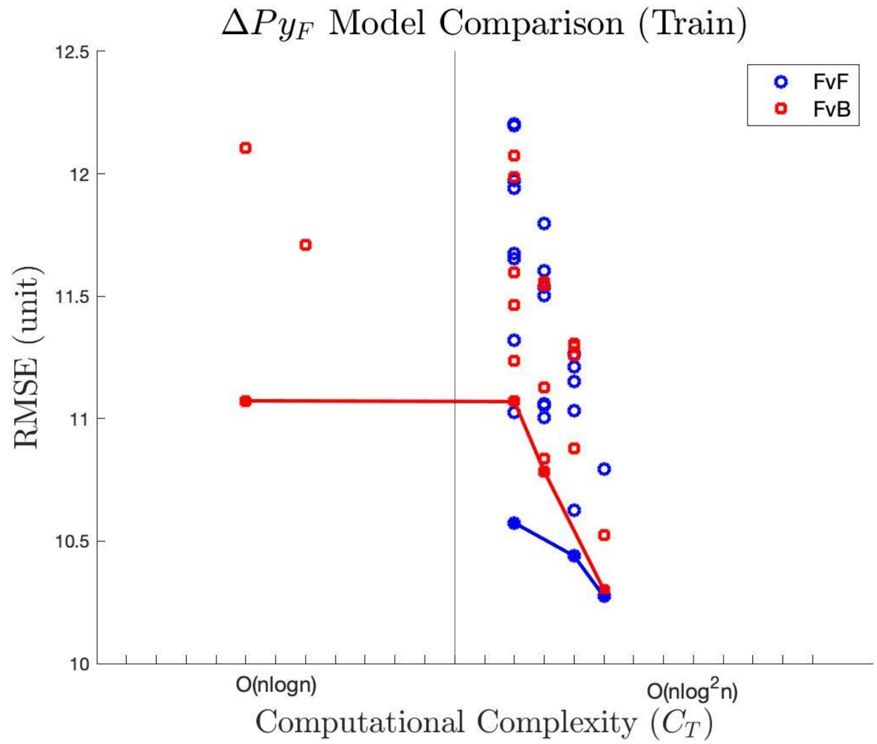


(a)

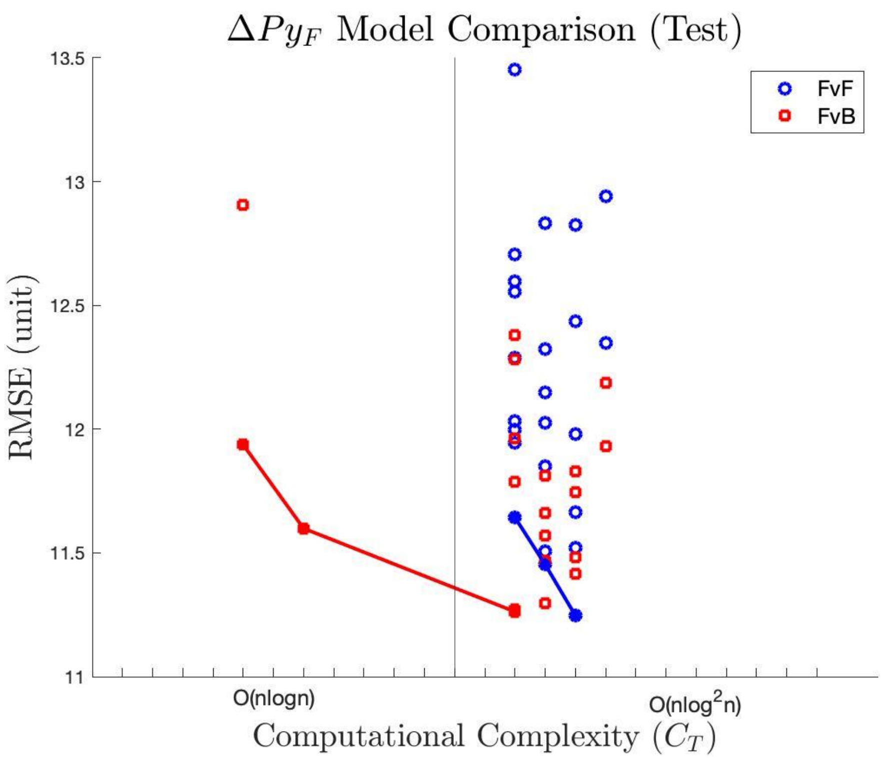


(b)

Figure 63 – Pareto optimal flock x-displacement models due to interaction and re-stabilization (a) training data (b) test data

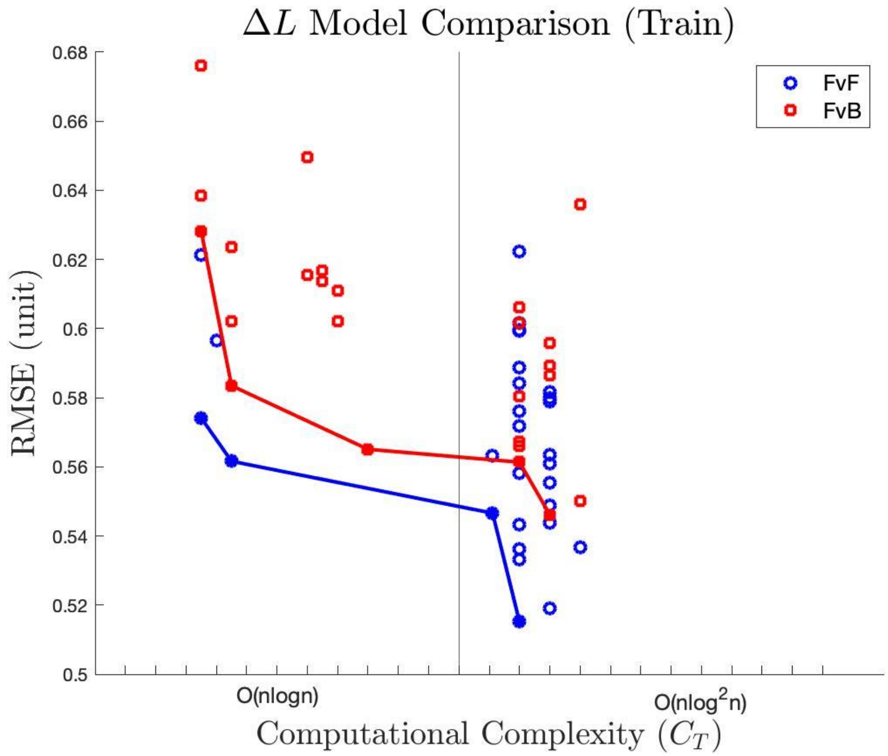


(a)

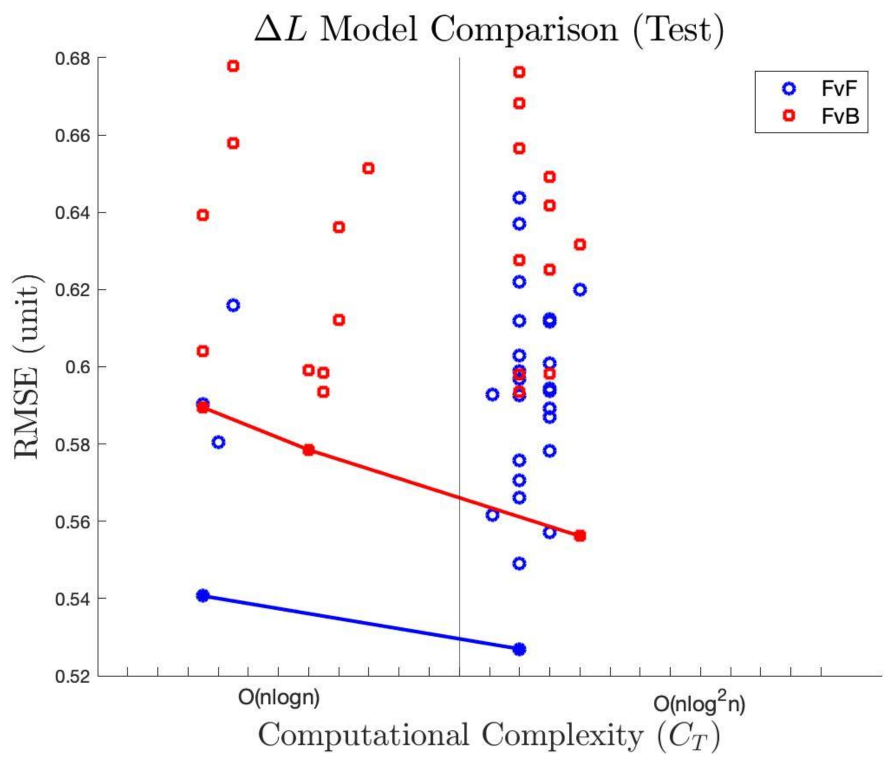


(b)

Figure 64 – Pareto optimal flock y-displacement models due to interaction and re-stabilization (a) training data (b) test data

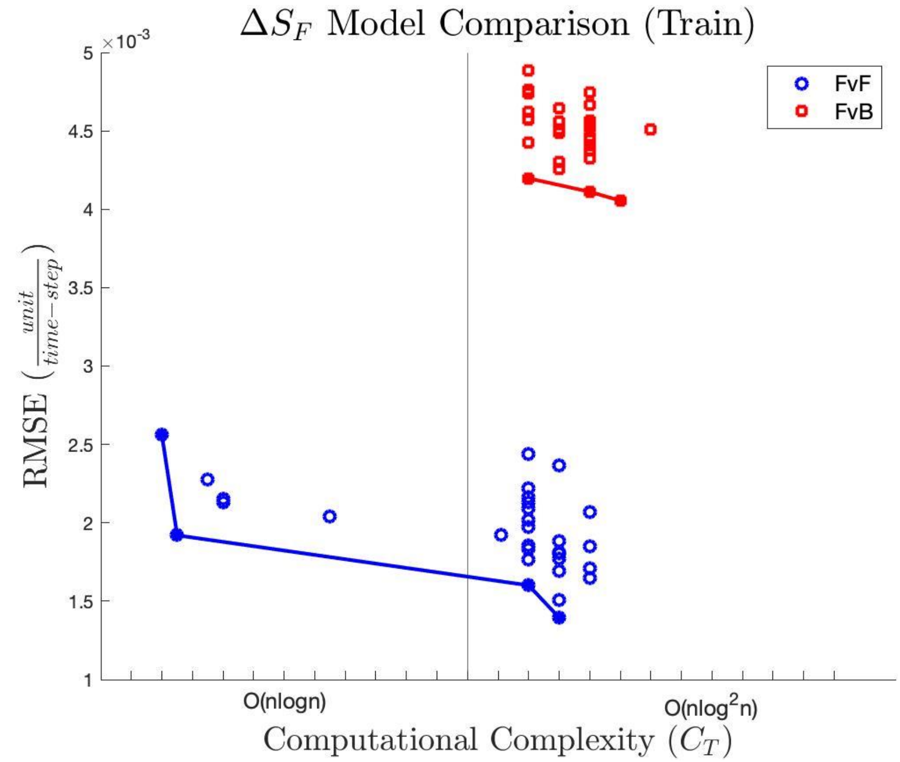


(a)

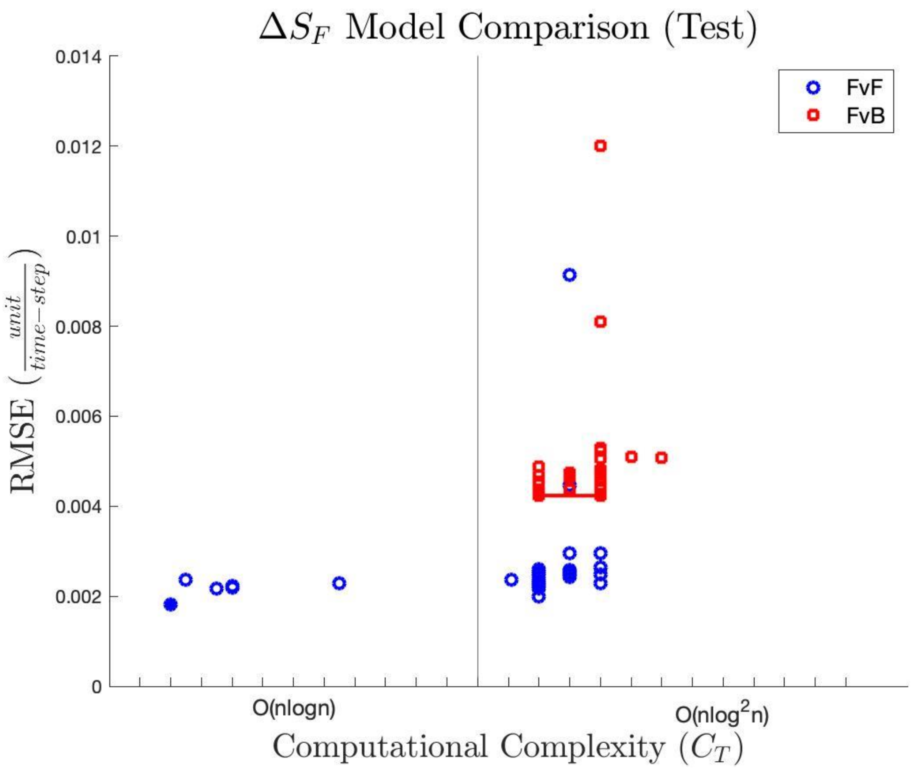


(b)

Figure 65 – Pareto optimal flock length models due to interaction and re-stabilization (a) training data (b) test data

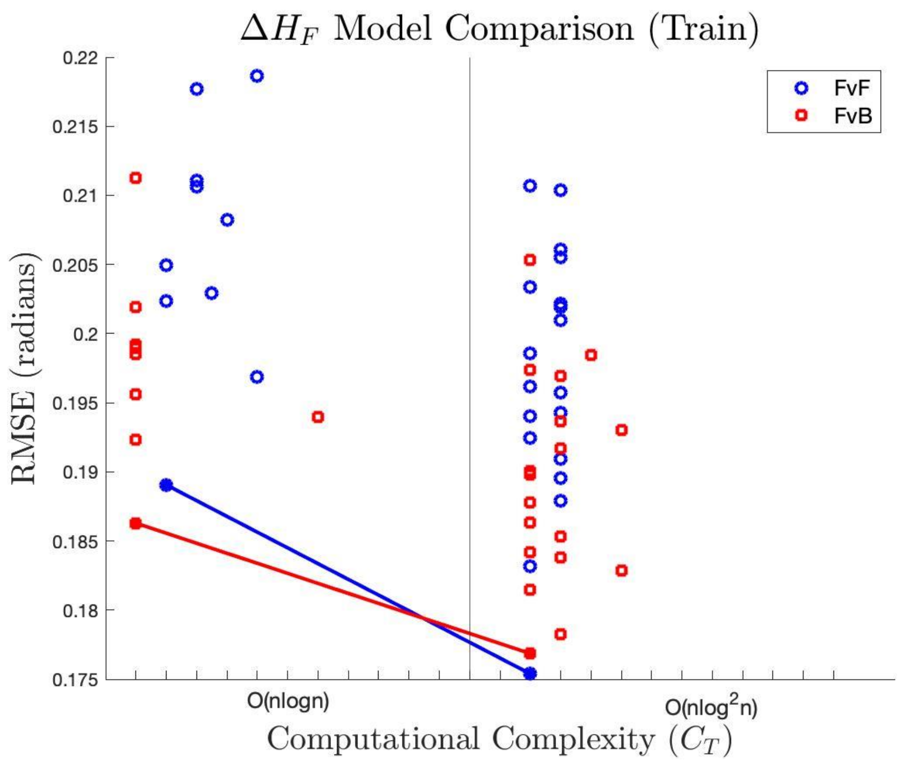


(a)

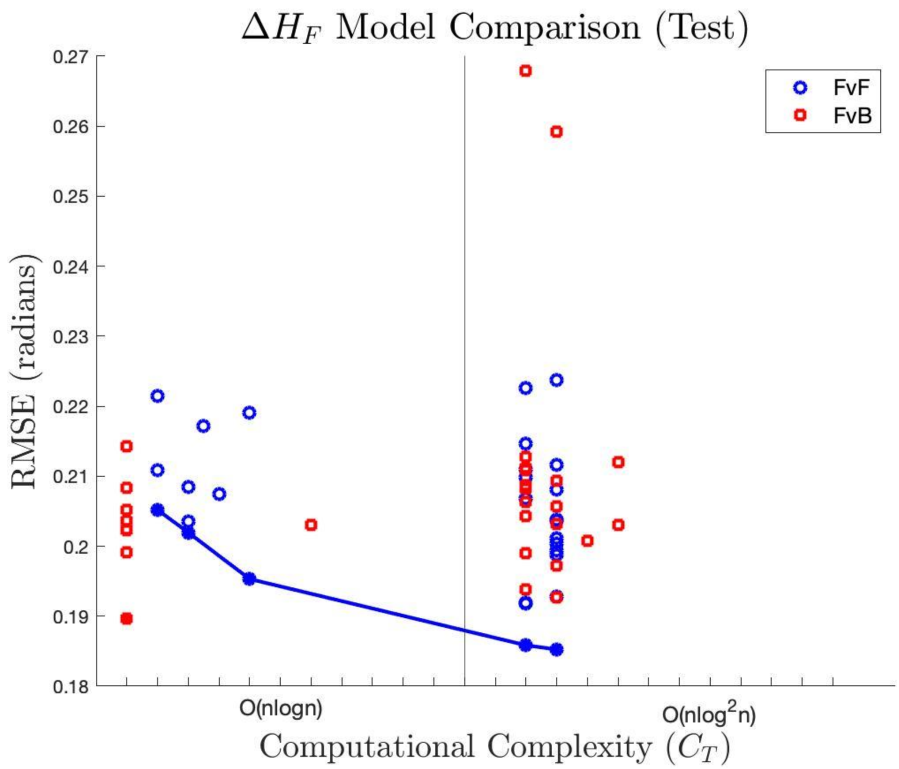


(b)

Figure 66 – Pareto optimal flock speed models due to interaction and re-stabilization (a) training data (b) test data



(a)



(b)

Figure 67 – Pareto optimal flock heading models due to interaction and re-stabilization (a) training data (b) test data

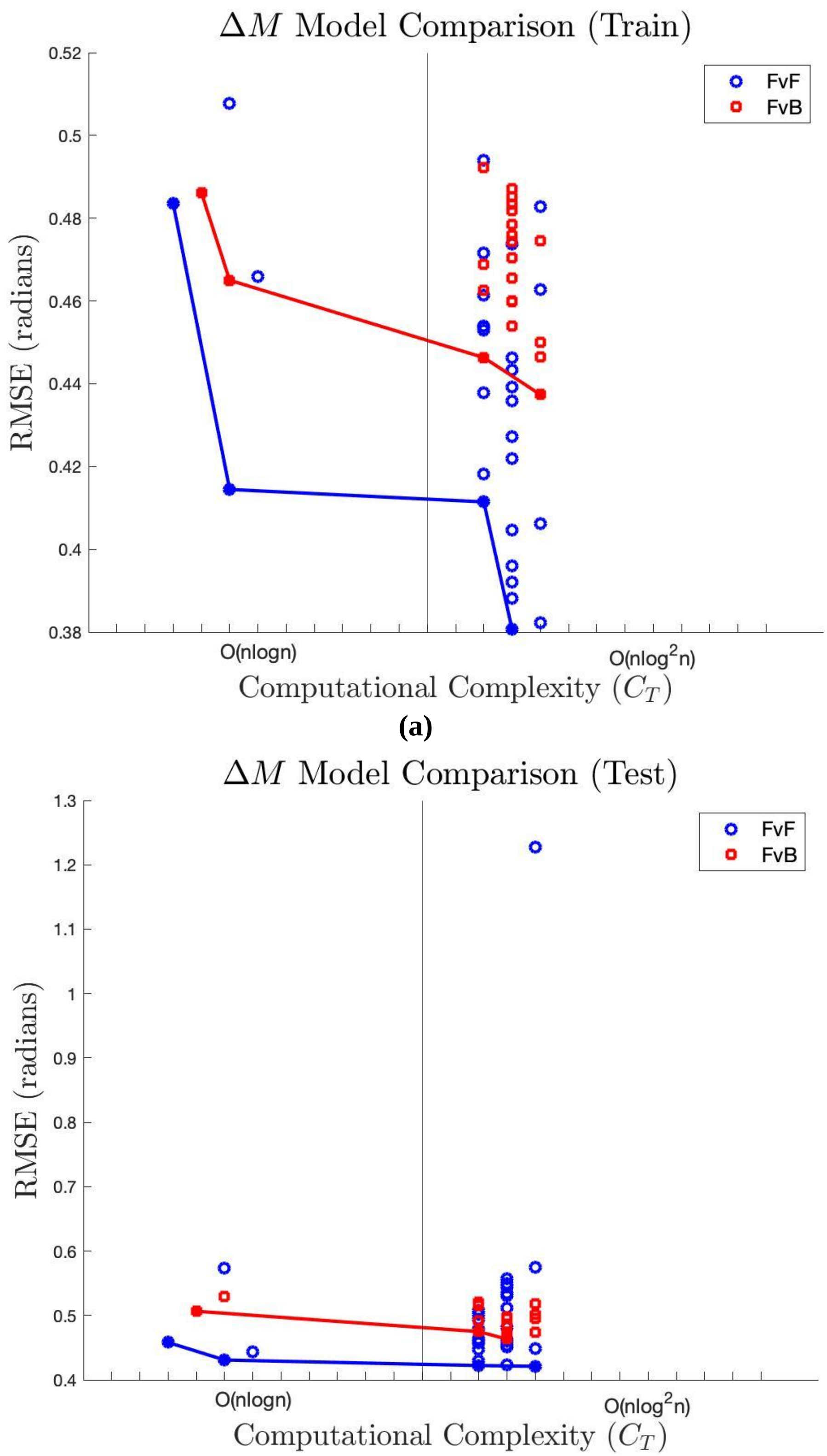


Figure 68 - Pareto optimal flock slope models due to interaction and re-stabilization (a) training data (b) test data

6.2.5 *Full Time-Series Data (H2 property interchangeability)*

The previous phase (Section 6.2.4) begins when the flock first begins to interact, and ends immediately once the flock has returned to a stable configuration. The time interval presented here, however, is the full time series (iteration 1 to 400). The key difference between these two is that, at iteration #1, the flocks are all stable, and so the heading and speed of the flock is numerically equal to the heading and speed of its boids. This presents the opportunity to (incorrectly) treat the variables as though they were interchangeable (the second test discussed in Section 5.5.2), thereby testing the numerical criteria's ability to produce a false positive.

Consider some of the Pareto Optimal equations for the heading of the flocks (see Figure 72b). The equations corresponding to the O(nlogn) category models are:

FvF: $$H_{F1,f} - H_{F1,0} = -0.11 + 0.036\left|n_2 H_{F1,0} - M_{F1,0} + H_{F2,0}\right| \tag{41}$$

FvB: $$H_{F1,f} - H_{F1,0} = 0.159 - 0.037\left|n_1 H_{2,0} - M_{F1,0} + H_{F1,0}\right| \tag{42}$$

FvB: (sim) $$\left.\begin{aligned} H_{F1,f} - H_{F1,0} &= -0.22 + 0.104\left|\left(H_{2,0} + H_{F1,0}\right) - \left(M_{F1,0} - H_{F1,0}\right)\right| \\ H_{F1,f} - H_{F1,0} &= -0.22 + 0.104\left|H_{2,0} + 2H_{F1,0} - M_{F1,0}\right| \end{aligned}\right\} \tag{43}$$

Three pieces of information stand out right away. First, these equations are nearly identical to the heading equations found during the interaction phase in Section 6.2.2. Second, these equations suffer from the interchangeability fallacy in discussed in Section 5.5.2. Third, there is a simplification for Eq. (43) that raises another important consideration for the calculation of computational complexity used in this thesis.

First, on the subject of the similarity, note that the only difference between Eq. (41) and Eq. (26) are the coefficients, and even then, the difference between coefficients is less than a factor of 25%. The same is true for Eq. (42) and Eq. (27). A FvF equation like Eq. (41) appeared in the Pareto Front of the interaction/re-stabilization data (Section 6.2.4), but was not discussed because the Pareto Front was dominated by the FvB model. The FvB Pareto Front did *not* contain an equation like Eq. (42), nor did SISSO produce a stand-alone equation similar to Eq. (43).[301] These equations did not appear in the SISSO results for re-stabilization (Section 6.2.3). So it seems that flock-level heading during any time interval containing the interaction phase can be somewhat well represented by this equation. It also seems that the equation outperforms the FvB models over the time intervals that are *not* appropriate for the numerical criteria. To put it in layman's terms, it would seem that this equation is very good at being wrong (or at least very good at being counter-intuitive). However, it is too early to draw any conclusions from this observation.

Second, these equations suffer from the interchangeability fallacy. Recall from Eq. (21) obtained in the stable time interval (Section 6.2.1), that one can write,

$$\begin{cases} H_{F1} = H_i \\ H_{F2} = H_j \\ H_{F1} = \mathrm{c}H_{F2} \end{cases} \tag{44}$$

provided that the headings are constant throughout the entire time interval.[302] Suppose, now, that the simulation domain was infinite, rather than finite and periodic (for ease of exposition). Eq. (44) would be valid for all time before the interaction, and for all time after

[301] Eq. (43) did appear as a term in a larger non-linear equation.
[302] For clarity: H_i is the heading of any boid contained in Flock 1. The heading of Flock 1 is given by H_{F1}.

the re-stabilization except that each time interval would have a different constant as shown in the following equation,

$$\begin{cases} H_{F1} = H_i \\ H_{F2} = H_j \\ H_{F1} = c_0 H_{F2} \; \forall \; t < t_I \\ H_{F1} = c_f H_{F2} \; \forall \; t > t_R \end{cases} \tag{45}$$

where t is time, the subscript $_I$ refers to the beginning of the interaction time interval, and the subscript $_R$ refers to the end of the re-stabilization time interval, and the subscripts $_0$, $_f$ simply distinguish the two constants. What all this means is that Eq. (41) shown below,

FvF: $$H_{F1,f} - H_{F1,0} = -0.11 + 0.036\left|n_2 H_{F1,0} - M_{F1,0} + H_{F2,0}\right|$$

can be replaced with,

BvF: $$H_{1,f} - H_{1,0} = -0.11 + 0.036\left|n_2 H_{F1,0} - M_{F1,0} + H_{F2,0}\right| \tag{46}$$

where H_1 is understood to mean the heading of a boid contained in Flock 1. In fact, any substitution between boid and flock could be performed throughout the equation.

BvB: $$H_{1,f} - H_{1,0} = -0.11 + 0.036\left|n_2 H_{1,0} - M_{F1,0} + H_{2,0}\right| \tag{47}$$

FvB*: $$H_{F1,f} - H_{F1,0} = -0.11 + 0.036\left|n_2 H_{1,0} - M_{F1,0} + H_{2,0}\right| \tag{48}$$

Based solely on the quantitative data, Eq. (41) and (46) – (48) have the same *RMSE* and C_T. Note the result for FvB*. In this equation, all inputs that could be replaced were replaced. The slope cannot be replaced by a boid-level property. Therefore, while it is possible to replace the dependent variable (under the basic premise of the interchangeability fallacy), it is not always possible to replace the input variables. Therefore, the numerical criteria were not fooled in this case. This confirms, however, that the selection of time interval, and the presence of distinct properties in the data set, is crucial to a proper implementation of an emergent behavior detection method.

Third, notice the simplification of Eq. (43). SISSO did not simplify the commutative terms, resulting in an equation where a variable is multiplied by an integer coefficient. In the multi-precision framework adopted for this thesis, this extra multiplication operation should increase the O(nlogn) complexity of the model. However, integer multiplication is really just repeated addition. Therefore, it is actually an increase in the O(n) complexity of the model, which is dramatically different. Furthermore, note that Eq. (41) – (42) include multiplications between heading and the variable n, which is the number of boids in the flock. Clearly, n is an integer. Regardless of how they are written (as multiplications between a real number and an integer, or multiple additions of real numbers), it turns out that Eq. (41) – (43) all have the same C_T. This unforeseen circumstance makes clear that estimating the complexity of a model is an inescapably nuanced process that demands at least some human judgment if not more research. Summarizing these observations about C_T, it seems that Kolmogorov complexity's fundamental ambiguity stems from not being able to ever find the shortest possible program to produce a desired output. The fundamental ambiguity in the complexity of finite-precision arithmetic comes from the large number of candidate algorithms that can compute the same result, and the various statistics involved in computing their performance (there is no way to decide that one metric is better than the other for all problems).[303] Finally, the fundamental ambiguity in the complexity of multi-precision arithmetic (used here) comes from the nature of mathematical representation and the fact that there are arbitrarily many ways to write the same expression, in addition to the ambiguity introduced by the linear speed-up theorem, and the inability to ever implement a galactic algorithm. In other words, when computing C_T one must choose between three inescapable constraints: (1 – arbitrary programs and Kolmogorov complexity) not knowing the best possible algorithm, and therefore, not knowing the exact complexity of that algorithm, (2 – finite precision arithmetic via any number of algorithms) knowing the algorithms and their complexities, but being unable to decide between them because they always trade one kind of performance for another, or (3 – multi-precision arithmetic via galactic algorithm) knowing the best possible algorithm, but being unable to ever implement it. These are deep waters. While each of these approaches exist because they are somehow practical, Kolmogorov complexity is clearly the most rigorously defined. The reader interested in adapting any of these methods is advised to pick the one they understand the most and examine the results of that approach with great skepticism. To this author's knowledge, this topic is not settled in the literature.

Now briefly consider the models for length. All of the FvB models in Figure 71b build off of the same term. This is typical for a single SISSO output file (though not guaranteed to happen), but less common for an entire Pareto Front.

[303] Recall the discussions in Section 5.1.3 and the Appendix.

FvB: $$L_{F1,f} - L_{F1,0} = 0.28 - 1.122\frac{L_{F1,0}}{n_1^2} \tag{49}$$

FvB: $$L_{F1,f} - L_{F1,0} = 0.20 - 29.96\frac{L_{F1,0}}{D_0 n_1^2} \tag{50}$$

FvB: $$L_{F1,f} - L_{F1,0} = 0.29 - 31.06\frac{L_{F1,0}}{D_0 n_1^2} - 0.657\frac{\cos\left(M_{F1,0}\right)}{H_{2,0} - H_{F1,0}} \tag{51}$$

FvB: $$\begin{aligned} L_{F1,f} - L_{F1,0} = -0.24 - 30.73\frac{L_{F1,0}}{D_0 n_1^2} - 0.003\frac{\exp\left(M_{F1,0}\right)}{H_{2,0} - H_{F1,0}} \\ -0.003\left(H_{2,0} + M_{F1,0}\right)\sin\left(M_{F1,0}\right) \end{aligned} \tag{52}$$

In Eq. (49) - (52), the variable D_0 represents the distance between the centroid of the flock and the location of the boid, and exp represents the well-known exponential function typically expressed as e^x. Note that Eq. (49) does not qualify as an interaction equation, but it is clear from the complete set of results that this term plays an important role in the accuracy of the regression. The fact that the properties of the flock have such a large impact on accuracy raises the immediate concern that perhaps all of the models built on it are erroneous. The next major improvement in accuracy comes from the introduction of the distance term, and that eliminates any doubt that the equations are unreliable because the initial positions of the flocks are arbitrary.[304]

A more interesting length equation is one of the FvF Pareto Optimal models:

[304] Recall that the decision to omit final values was made in order to avoid systems of coupled equations. Examining that possibility is left for future work.

FvF: $$L_{F1,f} - L_{F1,0} = 0.181 - 0.173\frac{L_{F1,0}M_{F2,0}}{n_1^2} \quad (53)$$

This equation will be discussed in further detail in Section 6.2.6. For now, it suffices to say that the $M_{F2,0}$ term undermines the credibility of this equation because a flock with zero slope must induce a constant change in length, which is unconvincing.

Speed data is not given for this time interval because it is constant at the beginning and end of the simulation. This selection of start and endpoint completely mask the interactions that alter the flock's speed. This highlights the importance of the selection of time interval, and suggests that this time interval is inappropriate. This is also an example of the "model not existing" during a particular time interval (see Appendix).

The slope FvF models are weakly dominated by the FvB models, however, the FvB model that causes this is invalid.

FvB: $$M_{F1,f} - M_{F1,0} = 0.179 - 0.00005\frac{\left(M_{F1,0}\right)^6}{H_{2,0}+H_{F1,0}} \quad (54)$$

This equation claims that if the initial slope of the flock is zero, then the final slope must be 0.179 radians. This is too strict to accept. Thus, the actual result for this time interval is that the FvF models dominate the FvB models. For the sake of consistency, however, the discussion in Section 6.2.6 will begin with the results as they appear in the various Pareto Front figures. Finally, both position variables show a clear preference for FvF models. All of the position models contain transcendental terms or cube roots, suggesting several are over fit. This will be compared to the results from other intervals in Section 6.2.6.

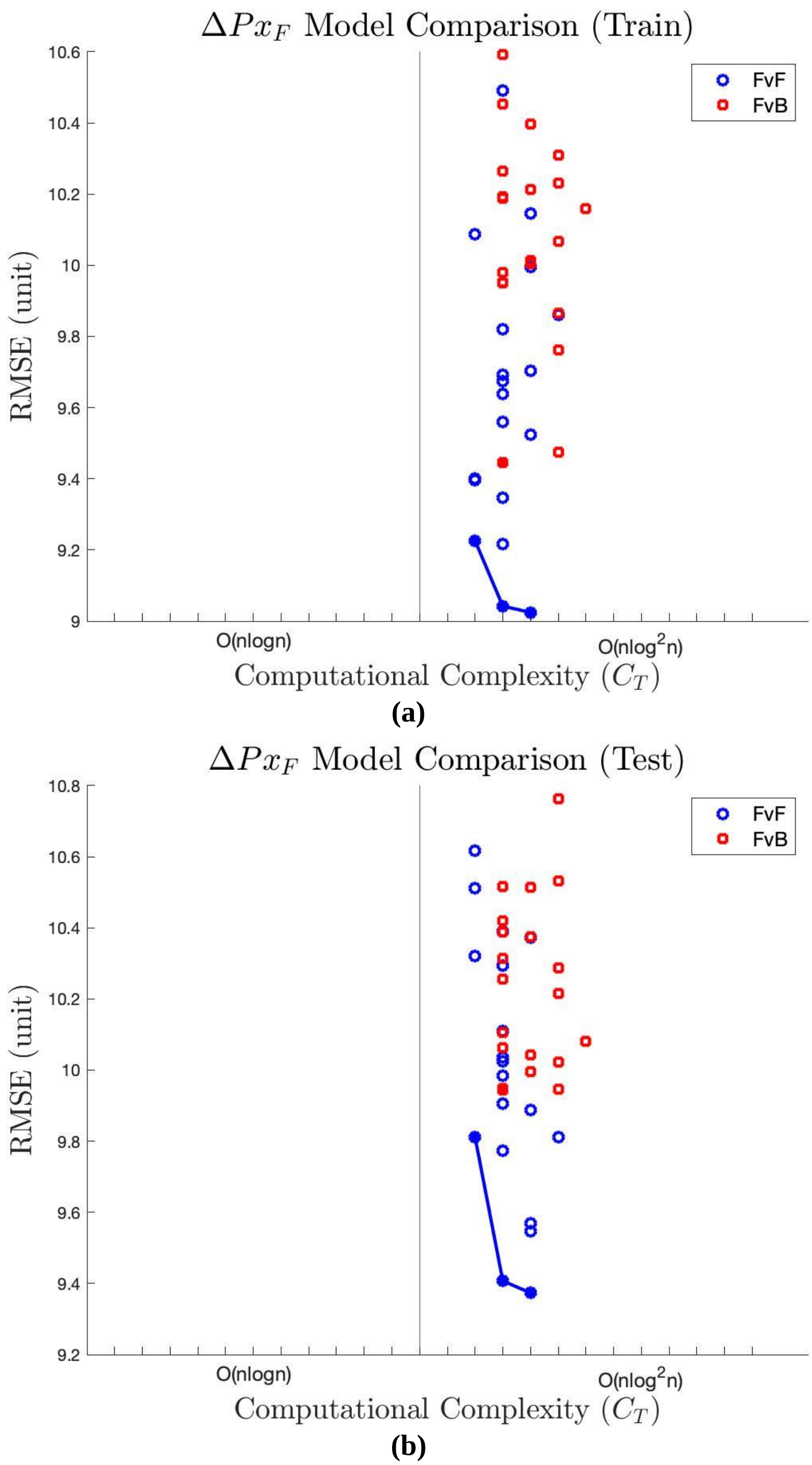


Figure 69 – Pareto optimal flock x-displacement models over full time interval (a) training data (b) test data

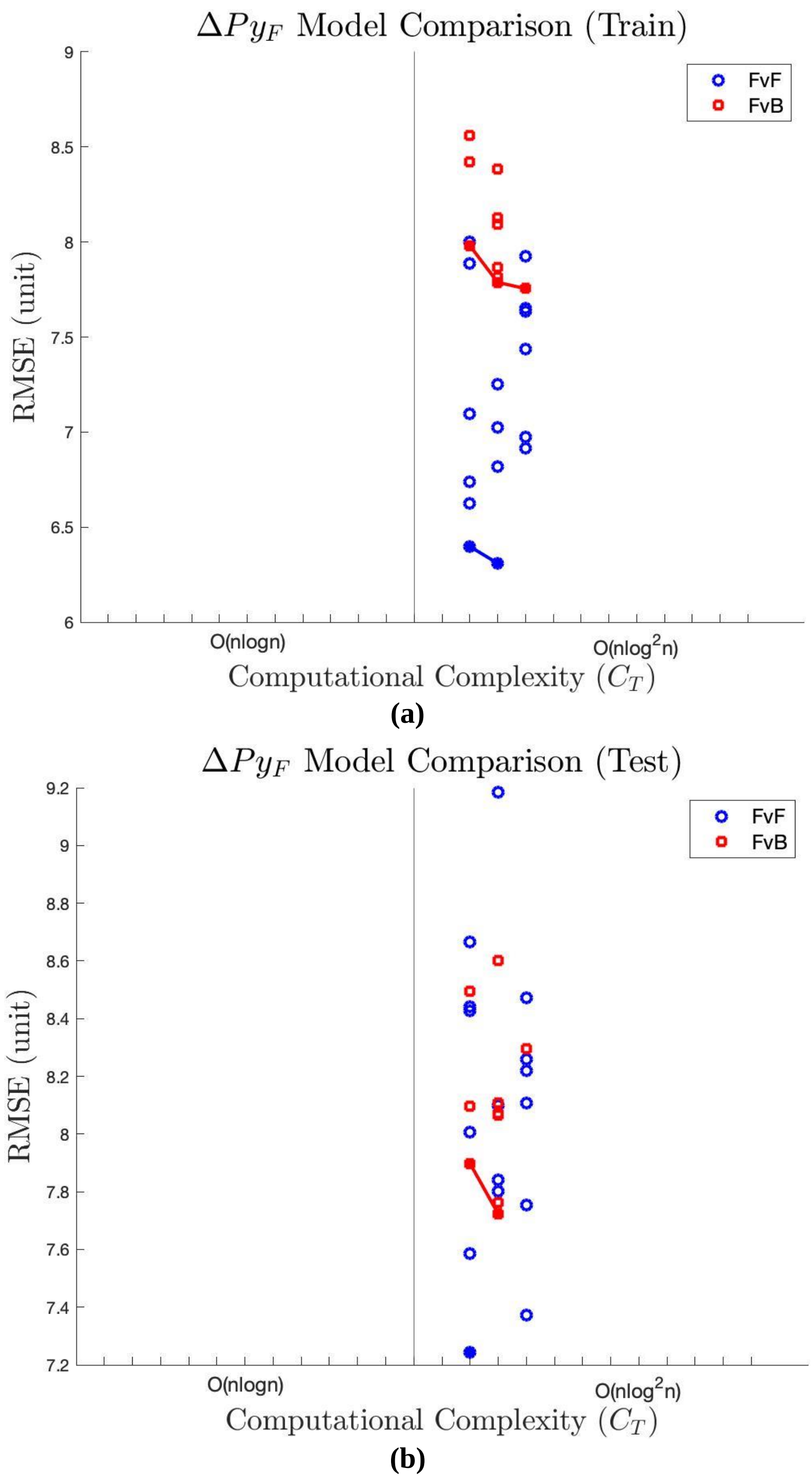
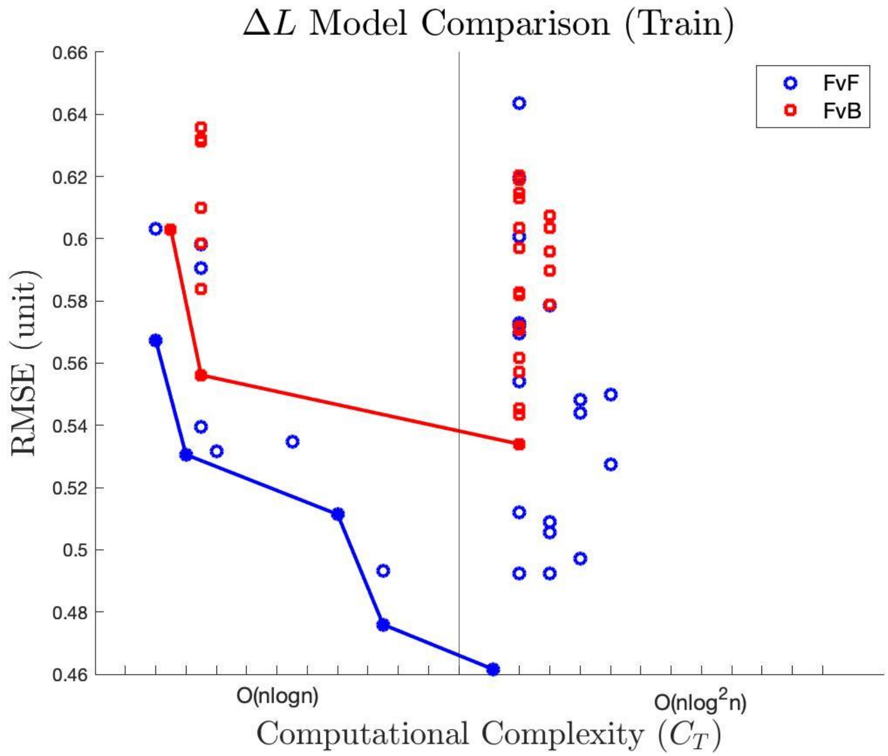
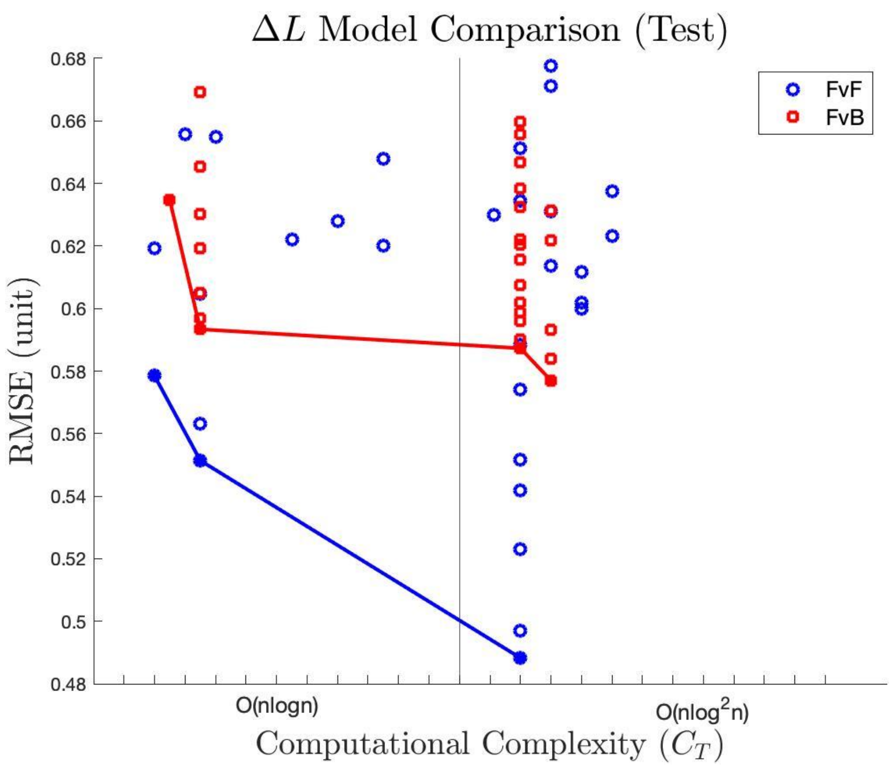


Figure 70 – Pareto optimal flock y-displacement models over full time interval (a) training data (b) test data



(a)



(b)

Figure 71 – Pareto optimal flock length models over full time interval (a) training data (b) test data

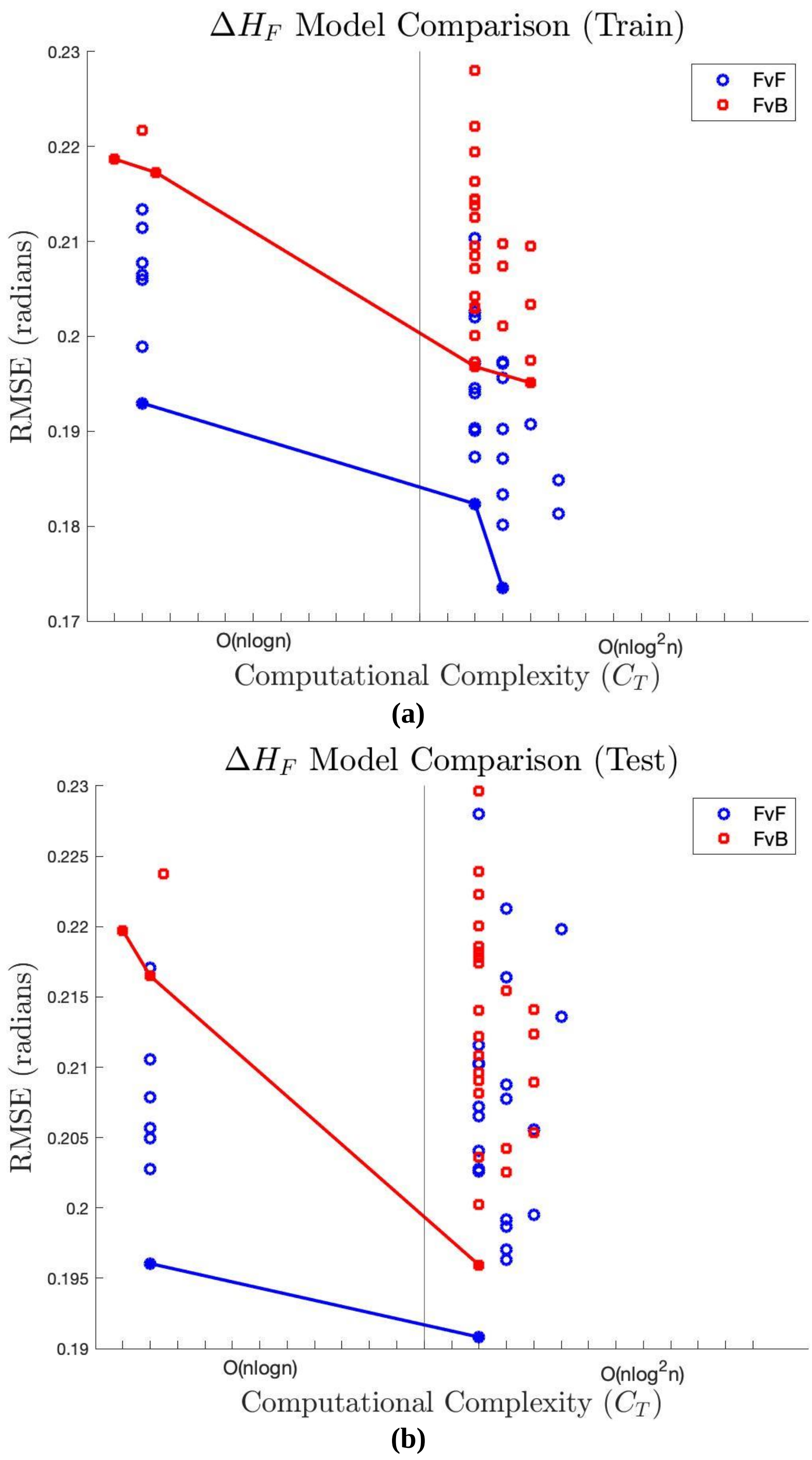
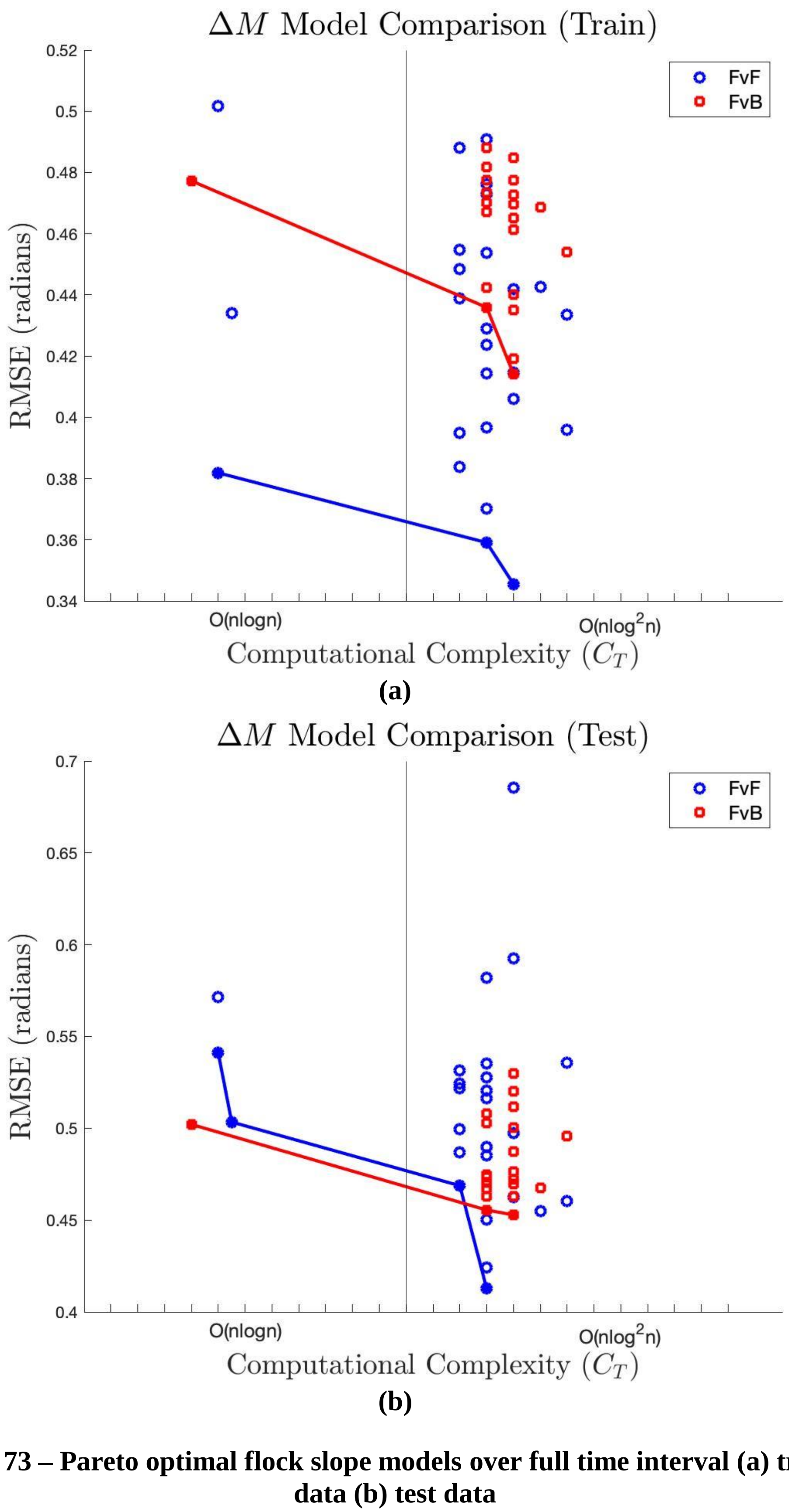


Figure 72 – Pareto optimal flock heading models over full time interval (a) training data (b) test data



(a)

(b)

Figure 73 – Pareto optimal flock slope models over full time interval (a) training data (b) test data

6.2.6 *Critical Analysis of H2-related results*

The results of the Section 6.2.1 – 6.2.5 are summarized in Table 8 below. Position variables were omitted from the list of candidate *inputs* in order to mitigate the issue of confounding outputs with upward-caused behavior (i.e. heavily coupled equations). In order to focus the reader's attention and facilitate assimilating the significance of these results, the narrative of this section will be presented in a question/answer format.

Table 8 – Summary of Pareto front comparisons for experiments in Section 6.2

Interval / Property	Stable / Independent (×,-)	Interaction (×,-,+)	Re-stabilization (×,-)	Interact / Re-stab. (×,-,+)	Full Time Interval (×,-,+)
ΔPx_F	-	-	+	-	+
ΔPy_F	-	×	-	-	+
ΔL	-	-	+	+	+
ΔS_F	-	+	+	+	-
ΔH_F	-	×	+	-	+
ΔM	-	-	+	+	-

What do the columns of Table 8 reveal about the numerical criteria? To simplify the notation slightly, the following acronyms will be used: Stable/Independent (SI), Interaction (IO for "interaction only"), Re-stabilization (RO), Interact/Re-stab (IR), and Full Time Interval (FT). The SI data supports that the numerical criteria are sufficient conditions since they indicate no emergent behavior over a time frame in which no interaction occurred. The RO data, however, shows the opposite, thereby falsifying Hypothesis 2 that the numerical criteria are sufficient conditions for identifying emergent behavior. The only behavior occurring during RO is intra-flock interactions as the boids attempt to re-stabilize their flock. This is an example of self-organization, not emergence, and yet the criteria would seem to suggest that every property other than Py_F corresponds to an emergent behavior. This indicates that the numerical criteria are sensitive to nonlinear changes in the values of properties, and cannot be applied blindly to just any time interval.

The IO, IR, and FT data cannot, in and of themselves, falsify Hypothesis 2, but the trends across these columns are instructive. First, no single property has a + in all three columns, nor is there a pattern of +/-/× for any property. Second, each of three columns indicate that no more than 4 properties are an emergent behavior. Whatever their source of error may be, the distinction being drawn is more subtle than merely "constant/linear versus nonlinear," as it might seem from comparing SI to RO (a small consolation).

Do any SISSO models recur in multiple time intervals, and if so, what is their significance? This section will focus on the Pareto Fronts obtained for the test data sets, and will discuss trends in the forms of the models contained in those Pareto Fronts for each variable. The x-coordinate (Px) in the IO, IR, and FT time intervals has FvB and FvF Pareto Optimal models that contain $\sin(H_{F1,0})$[305] as part of the leading term in one or more points, and it is in the numerator as one would expect. All models for the FT time interval contain the sine term. Neither of the two Pareto Optimal models in the RO time interval contain

[305] Recall that the NetLogo coordinate system is not the standard right-handed system.

$\sin(H_{F1,0})$. Regarding the y-coordinate (Py) in the IO, IR, and FT time intervals, none of the FvF models contain speed as an input variable, whereas in RO all of the FvF models contain $(S_{F1,0})^6$ in the numerator of the lead term. Models for speed (S) of the form used in this thesis can only be generated in the IO, RO, and IR time intervals. All of the FvB and FvF models obtained for speed in the RO time interval are univariate functions of the initial speed (whether the initial speed of the flock or the opposing boid). All of the FvF and FvB models for IO, and IR, on the other hand, are multivariate functions (only some contain initial speeds). Thus, Px, Py, and S all exhibit clear differences between the models generated for time intervals with interactions (IO, IR, FT), and the time interval lacking interactions (RO).

The flock heading (H) shows a striking trend within the Pareto Optimal models for IO, IR, and FT. Specifically, the following equation re-appears in each time series (with only slightly different coefficients),

FvF:
$$H_{F1,f} - H_{F1,0} = c_0 + c_1\left|n_2 H_{F1,0} - M_{F1,0} + H_{F2,0}\right| \qquad (55)$$
$$c_0 \sim -0.109, \qquad c_1 \sim 0.036$$

This error of this equation within the context of the IR time interval is depicted in Figure 74 below. Figure 74 shows that although Eq. (55) is not yet accurate enough to be useful in an engineering context, the errors are spread somewhat evenly across the range of prediction (relative to the other models obtained). For readers that might be inclined to dismiss this result as terrible, please carefully review Figure 75 in order to get a sense for just how pathological the SISSO results can truly be.

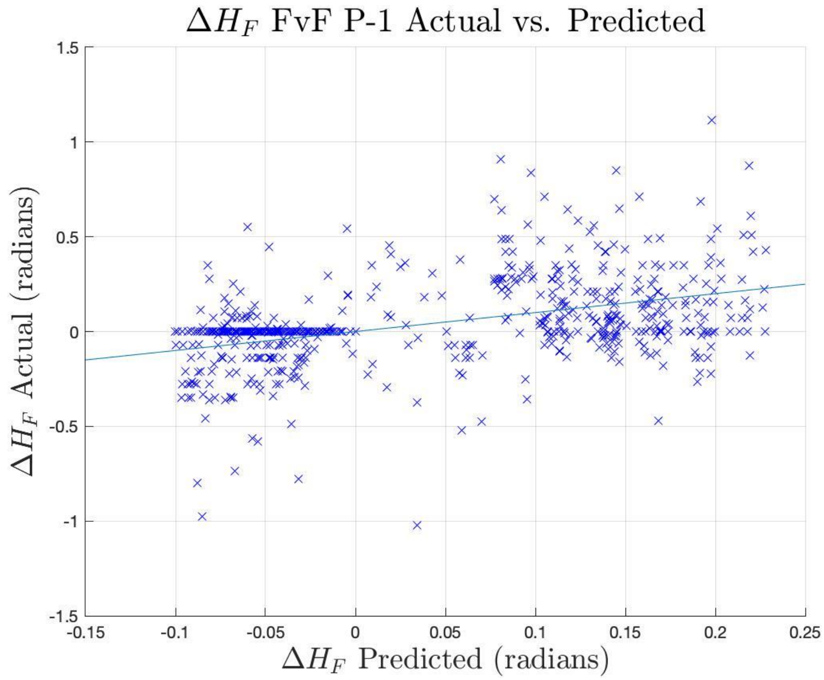


Figure 74 – Flock Heading Actual versus Predicted plot (interpolation and extrapolation data) for Eq. (55)

Figure 75 shows that there are many ways to be wrong while simultaneously obtaining a "reasonable" RMSE. Each of these images is a Pareto Optimal model either from the training data set, or both the training and test data sets.[306] Compared to those results, Figure 74 *suggests* that a useful, predictive model is within reach by building from the nonlinear terms and variables in Eq. (55).[307,308,309] Now compare the Predicted vs. Actual plots for the FvF models in the IR time interval to those of the FvF models in the RO time interval (shown in Figure 76 below).

[306] Generally there are fewer points in the test data Pareto Fronts because the interpolation points fail to extrapolate.
[307] The fact that the equation reappears is also suggestive of the same thing.
[308] Figure 75c is the closest in appearance to Figure 74, but still visibly worse. The two models are different.
[309] Recall that the objective of this thesis requires developing a method for dealing with emergent behavior exploitation. The question of "how do I know the exact equation" is a "model discovery" question that has been scoped out of this thesis. The goals of this thesis can be achieved without the exact model because the upward causation equations are known (this will be explored in CHAPTER 7).

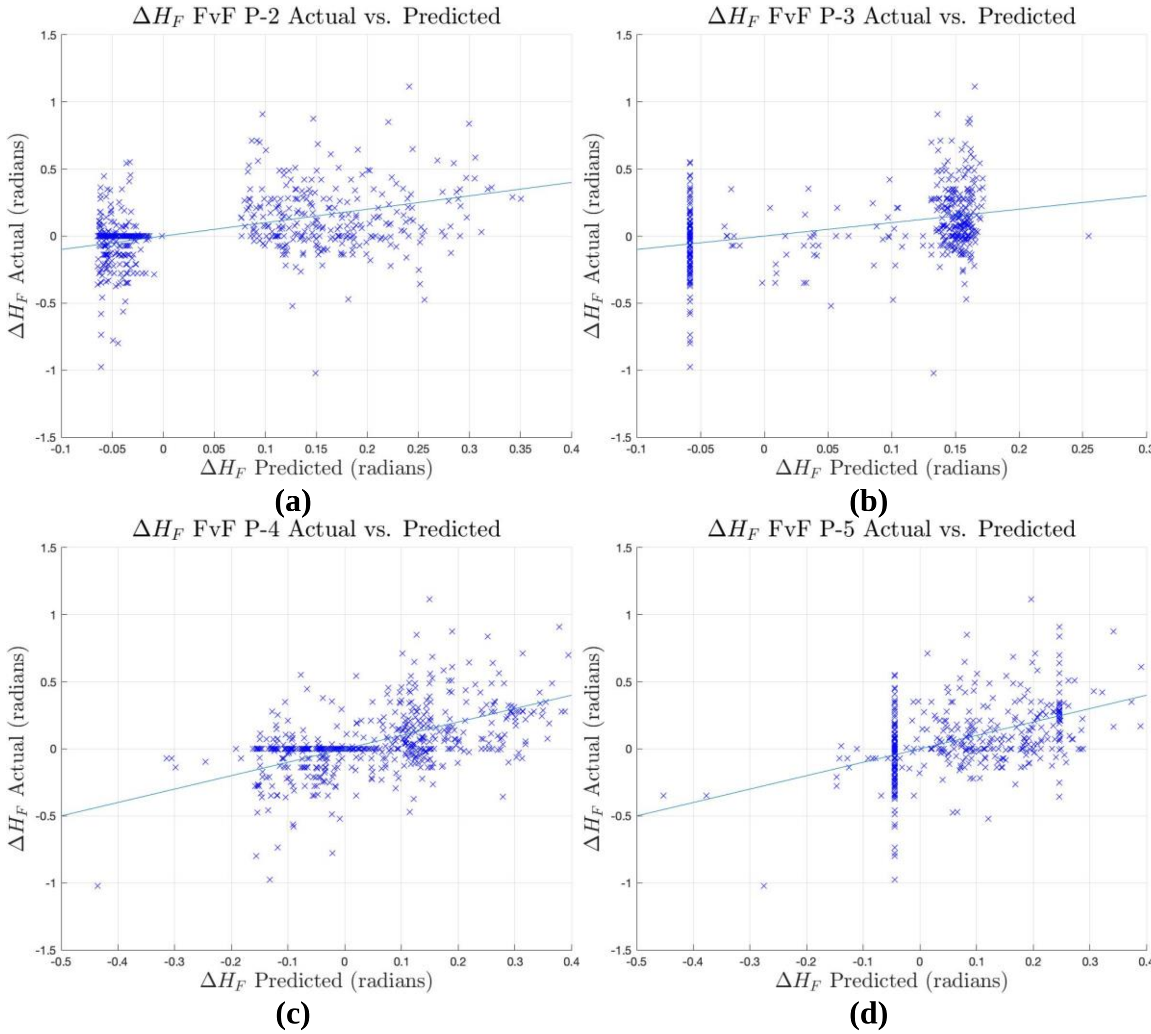
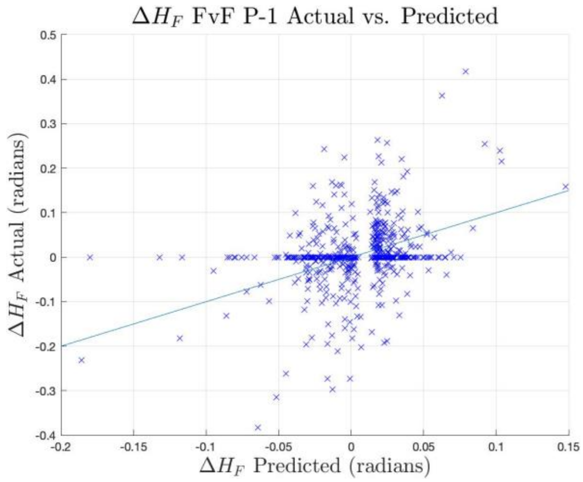


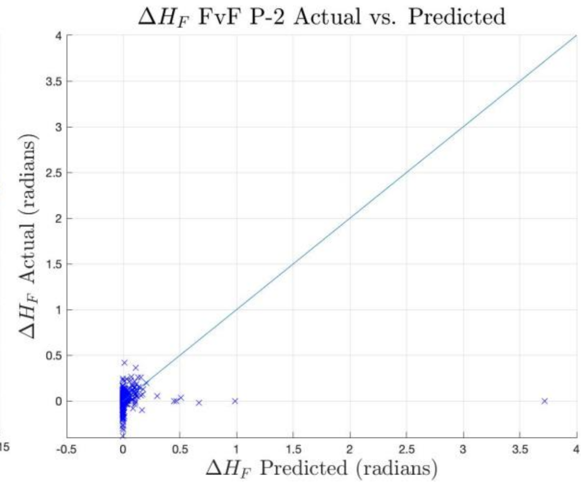
Figure 75 – Flock Heading Actual vs. Predicted plots (interpolation and extrapolation data) for IR Pareto Optimal models

The error distributions in these plots are clearly as bad, if not worse than the errors in Figure 74 - Figure 75. More importantly, the RO Pareto Optimal equations do *not* resemble Eq. (55), nor the models for the IR time interval in general. While other less common, or even unique, models also appear in the Pareto Fronts for IO, IR, and FT, those equations also do not resemble the results of the RO models.[310]

[310] Presenting the results here would unnecessarily clutter the document. It suffices to say that SISSO returned many different nonlinear forms.



(a)



(b)

Figure 76 – Flock Heading Actual vs. Predicted plots (interpolation and extrapolation data) for RO Pareto Optimal models

In other words, the best-case results for the RO time interval is as bad or worse than the results for the IR time interval, and contains models whose form is fundamentally different from those of the time intervals with interactions. The results for time intervals with interactions is of a quality altogether different than the results from the RO time interval.

In addition to the FvF models for heading, there is an FvB model that reappears in both the FT and IO time intervals (not IR):

FvB:
$$H_{F1,f} - H_{F1,0} = c_0 + c_1\left|n_1 H_{2,0} - M_{F1,0} + H_{F1,0}\right|$$
$$c_0 \sim 0.15, \qquad c_1 \sim -0.035 \tag{56}$$

This FvB equation also differs from its RO counterparts.

Finally, consider one of the Pareto Optimal models for length (L) in the FT and IR time intervals.

FvF:

$$L_{F1,f} - L_{F1,0} = c_0 + c_1 \frac{L_{F1,0} M_{F2,0}}{n_1^2} \quad (57)$$

$$c_0 \sim 0.195, \qquad c_1 \sim -0.185$$

This result is totally different from the RO Pareto Optimal RO models, all of which contain the same speed term raised to the 6th power.

FvF/FvB:

$$L_{F1,f} - L_{F1,0} = c_0 + c_1 S_{F1,0}^6 f(\theta) \ldots \quad (58)$$

Here, θ is a dummy variable, because the terms that follow are apparently arbitrary collections of nonlinear terms that enable the regression. In addition to consistently possessing the 6th order term, the coefficients across all of the RO equations vary by multiple orders of magnitude suggesting a very poor fit.

The key takeaway from these results is that for 5 out of the 6 variables[311] there is a clear trend toward one set of models (and one set of influential variables) in data sets that contain interactions versus a totally different trend in the models for the RO data set. Qualities that stand out in Pareto optimal models for RO does not appear in IO, IR, and FT, and vice versa. Sometimes that quality is the form of the equation, other times it is the selection of input variables to the model. The fact that different trends exist in the models generated from the two data sets is consistent with the expectation that the results for the RO and SI time intervals should be different from the results in the IO, IR, and FT time intervals. So while the numerical criteria in their current form fail to invalidate the models

[311] Slope has no remarkable similarities or differences in its results. This will be discussed later in this section.

for the RO time interval,[312] the results obtained by SISSO do reflect the different dynamics of those two situations. This can be used to improve the numerical criteria.

What do the trends in models across time intervals suggest about the implementation of the numerical criteria? Since this work takes the stance that the system of interest is a self-organized system, and since periods of stable, independent behavior do not qualify as time intervals for emergent behavior detection, the most appropriate time interval for the numerical criteria is the IR time frame.[313] However, this immediately suggests that the numerical criteria are only necessary conditions. In order to implement an emergent behavior detection scheme properly, it is clear that two additional pieces of information must be tracked: (a) the types and durations of system-level interactions, and (b) the time required for the perturbed system to re-stabilize. It may even be possible, in some special cases, to turn emergent behavior detection into an inverse design problem by examining the perturbations of the system and how they disrupt or change the periodic interactions between components.[314] For example, if the coefficients of the periodic function change, that may be used to infer the form of the system property and reverse-engineer the changes to the system property. At a minimum, perturbations of the system can be used as a preliminary indication of an interaction.

Is this the right approach for vector-valued properties? It is reassuring to see that SISSO inserted sin(H) into most Px equations and cos(H) into most Py equations. However, the position variables in Table 8 do not show a consistent pattern between the

[312] They "fail" to invalidate the RO model because the criteria do not call for discarding RO data. That is, they possess no mechanism for disqualifying data on the basis that it came from the wrong time interval.
[313] In a sense, the time period between intervals of stable, independent behavior.
[314] These are likely to be *very* simple cases only, but simple cases can also be important.

Pareto Optimal results of the x- and y-coordinates. In order to simplify the DoE for these experiments, at least one flock was always initialized with a heading of zero (i.e. an upward trajectory), and the opposing flock would fly towards it from a number of different trajectories. Perhaps the fact that a significant portion of data has the same initial behavior biased the results for the positions. In any case, more experiments are needed to determine if the numerical criteria are valid for vector-valued properties.

Why do the models for slope not have a distinguishable trend? Recall that the flocks are treated as though they were lines, so it was thought that two properties typically associated with lines would be good candidate emergent properties. However, the determination that the flock is a stable line is based on the initial and final configurations of the boids over the full time series. It was occasionally observed that the boids can move significantly during the flock-level interactions and subsequent re-stabilization. Therefore, there are cases where the idea of a linear flock strains credulity, which introduces a kind of error into the values of slope at the end of the IO, and beginning of the RO time intervals. This author sees those errors less as a kind of numerical inaccuracy,[315] and more as a subtle category fallacy. Although the boids can be arranged linearly, the flocks are not truly lines. So why does length exhibit a trend but not slope? Any two dimensional distribution of points can be assigned a characteristic length without introducing a fallacy (a circle has a diameter, a cloud of atoms has a mean free path, etc.), but the slope is only meaningful if the points are very nearly linear. Perhaps the reason the slope models did not have any kind

[315] The Matlab regression algorithm reliably produced the only reasonable value for slope possible. The issue is not that it was a "bad fit" but that the boids were not actually a line.

of pattern is because slope values lose their meaning faster than length values (i.e. the “line” abstraction only goes so far).

What to these results reveal about Hypothesis 2? As discussed before, a complete set of numerical criteria would require some form of interaction detection. And re-stabilization detection (for property change calculation).

Hypothesis 2 is falsified *in that the Numerical Criteria are not sufficient conditions.*

Furthermore, after observing that so many models have clear trends in one set of time intervals but not the other, it seems that this would be very useful and important information to include in an emergent behavior detection method. After all, the major distinctions came between time intervals containing an interaction, and those that did not. Since the emergent behaviors in this thesis must be functional, the appearance of different model types might suggest that the behavior being modelled is, in fact, functional. This is an opportunity for future work. This completes the experiment for Hypothesis 2.

6.2.7 H1 Falsification Test

Since Hypothesis 2 was falsified, this test will rely on a worst case scenario: the number of properties for the smallest possible flock will be compared to the largest number of properties found in any column of Table 8, which is the re-stabilization column (5 properties were “identified”). First recall the equations for the boid dependent variables,

$$\vec{x} = \vec{V}t + \overrightarrow{x_0} \tag{59}$$

$$\vec{V} = \begin{pmatrix} S \sin H \\ S \cos H \end{pmatrix} \tag{60}$$

To simplify notation, the velocity vector will be treated as the dependent variables rather than speed and heading. For a 2-boid linear flock there are eight dependent variables (two position coordinates, and two velocity components per boid): $C_S(M_0) = 8$. Once both boids form a line, the equations simplify to the following equations,

$$\overrightarrow{x_1} = \overrightarrow{V_1}t + \overrightarrow{x_{1,0}} \tag{61}$$

$$\overrightarrow{x_2} = \overrightarrow{V_1}t + \overrightarrow{x_{1,0}} + \vec{c} \tag{62}$$

The position of the second boid can be re-written using the velocity of the first, meaning six dependent variables remain, $C_S(M_R) = 6$. According to Hypothesis 1 the maximum number of emergent properties is 8 – 6 + 1 = 3, which is consistent with the IR results!

Hypothesis 1 is falsified *in this worst-case scenario.*

CHAPTER 7. ADVERSARIAL BOIDS CASE STUDY

As with CHAPTER 6, the pilots have only two interaction-dependent properties: heading and speed. Strictly speaking, the interactions are one-way (in large part because they are adversarial), but due to the large increase in vision cone angle, the need to avoid collisions, and the behaviors of attacking and avoiding attack, most one-way interactions will occur simultaneously and/or sequentially. A true two-way interaction would be something like the force of gravity between two massive objects. One important difference between this and Flocking Vee model is that the components will never all self-organize into a single entity because interactions between pilots disrupt formations much more often than not. This means that it is typically possible to apply the definition of an emergent behavior to the self-organized entities in this simulation. There may exist a case where the adversarial self-organized systems become locked in some kind of pattern (a form of stalemate), but that has not yet been observed in the data.

7.1 SO Detection

Due to the long time intervals of these simulations, the calculation of pairwise separation between pilots requires fully accounting for the fact that that the domain is a 101×101 torus with coordinates ranging from [-50.5,50.5) in each direction. A rigorous, but unnecessarily complicated approach, would be to map the square domain onto a torus, compute the shortest distance between pilots, and then map that distance back onto the square domain starting from the procedures described in [251] [252]. Fortunately, simpler procedures exist for the task at hand, which happened to be conveniently listed on a nonscholarly website [253].[316] The tiling procedure discussed on that website produced results with discontinuities, just like the naïve distance formula would, and underestimated the distance between pilots in some cases. The more accurate procedure is the simple equation,

$$\left.\begin{array}{c} D_x = min(x_1 - x_2, 101 - |x_1 - x_2|) \\ D_y = min(y_1 - y_2, 101 - |y_1 - y_2|) \\ L = \sqrt{D_x^2 + D_y^2} \end{array}\right\} \tag{63}$$

The reader can verify that the maximum possible separation returned by Eq. (63) corresponds to the distance between a pilot at the center of the square and a pilot at any corner of the square, which is the correct result. By plotting this distance equation over the course of the simulation along with the relative heading and vision information of the pilots, one can easily visually identify stable formations and pursuit systems.

For example, consider the full time series plot of pairwise distance and relative heading for two blue-team pilots, shown in Figure 77. This time series is much noisier than the inter-boid distance plot shown in Figure 27.[317] Nevertheless, as in Figure 27, the flat regions of the graph correspond to stable formations of pilots. One such region is highlighted in yellow and labelled "Stable" in Figure 77. Figure 77d shows a screenshot of the two blue pilots flying in the corresponding stable formation (the yellow arrow indicates

[316] The web address is too long to properly cite using MS Word's citation feature: https://answers.yahoo.com/question/index?qid=20070806182105AAXA7ob&guccounter=1&guce_referrer=aHR0cHM6Ly9kdWNrZHVja2dvLmNvbS8&guce_referrer_sig=AQAAAJm-edMjNxiTwyrU0tN3-MN2OtoQGXbdfnkcMOVflkP0lmDmjEJ5cMkBebSLoYCmFaFFrQEufQB9eIpmves4R6fD-mPyUsL-SQqMOTZDtSw8bklVFvaPQMe9NhS1w3lHpvcQyWDF7RYEwHE__xKHLrzoSGb2iOuL5nuKxIVqwcsq

[317] The time-series could be smoothed using a moving average, which would also make it amenable to analysis using ARIMA models and Granger Causality. The notion of a periodic moving average will not be explored in this thesis, but lends itself to dynamic structures, and should be studied as part of a more general definition of self-organization.

their trajectories). Four other regions in Figure 77 are highlighted in green, and one of those regions is labelled “Unstable.” The screenshot corresponding to the labelled region is given in Figure 77c. This configuration is unstable because the two pilots are accelerating along a curved path, in pursuit of a pair of red pilots. Most perturbations will cause this formation to break (e.g. collision avoidance, enemy fire, etc.). This 2-versus-2 pursuit system will not be studied, but is a candidate for emergent behavior detection. These unstable configurations are more difficult to identify visually, and require additional vision-cone information in order to detect algorithmically. Since the unstable formations are relatively very short lived (less than 100 iterations) the windowing procedure discussed in Section 5.2.2 should be supplemented with interaction-detection information in order to function efficiently. Note that three of the unstable configurations have a relative heading near zero and a relative distance near that of the stable configuration. Each of these turns out to be variations of a central theme (the so-called “Fighting Two” formation [43]). The fourth unstable configuration at the very end of the simulation is a Fighting Two that is in the process of forming up (the pilots are already close and the wingman is turning to follow the lead).

An example of a 1-versus-1 Pursuit System time series is depicted in Figure 78. One stable configuration (labelled similar to Figure 77) results in sinusoidal region in the distance plot and a flat region in the relative heading plot (see discussed in Section 5.2 and CHAPTER 3). These two pilots circle each other for a very long time, as depicted in Figure 78d, until finally another blue pilot intercepts the red pilot. This, too, is a precursor to the Thach Weave. The sine wave is caused by the attacker and target’s acceleration as one tries to fly into a favorable position while the other tries to cause its pursuer to overshoot.

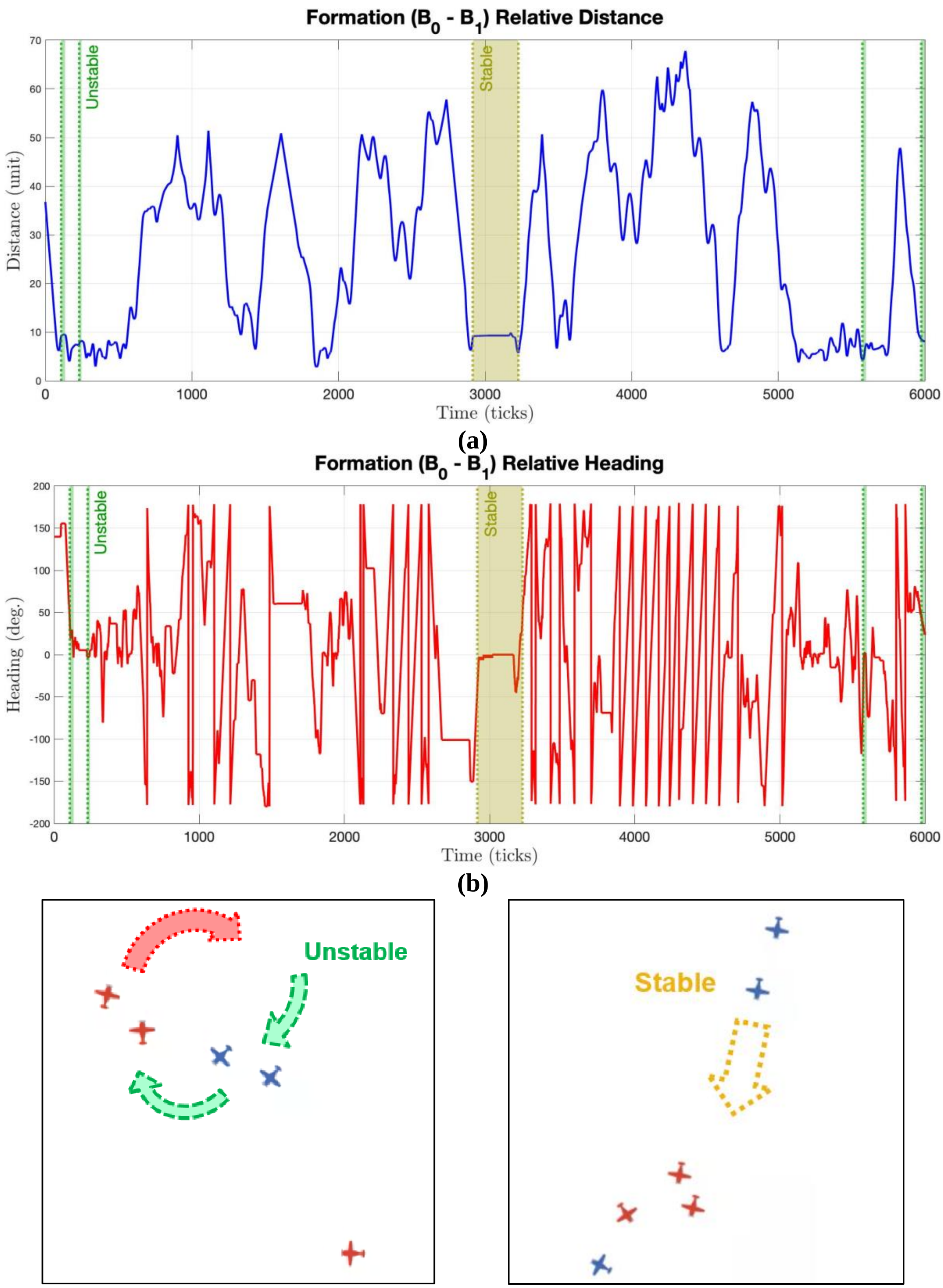


Figure 77 – Time series data corresponding to Adversarial Boids simulation (a) relative distance between blue pilot 0 and 1, (b) relative heading, (c,d) screenshots

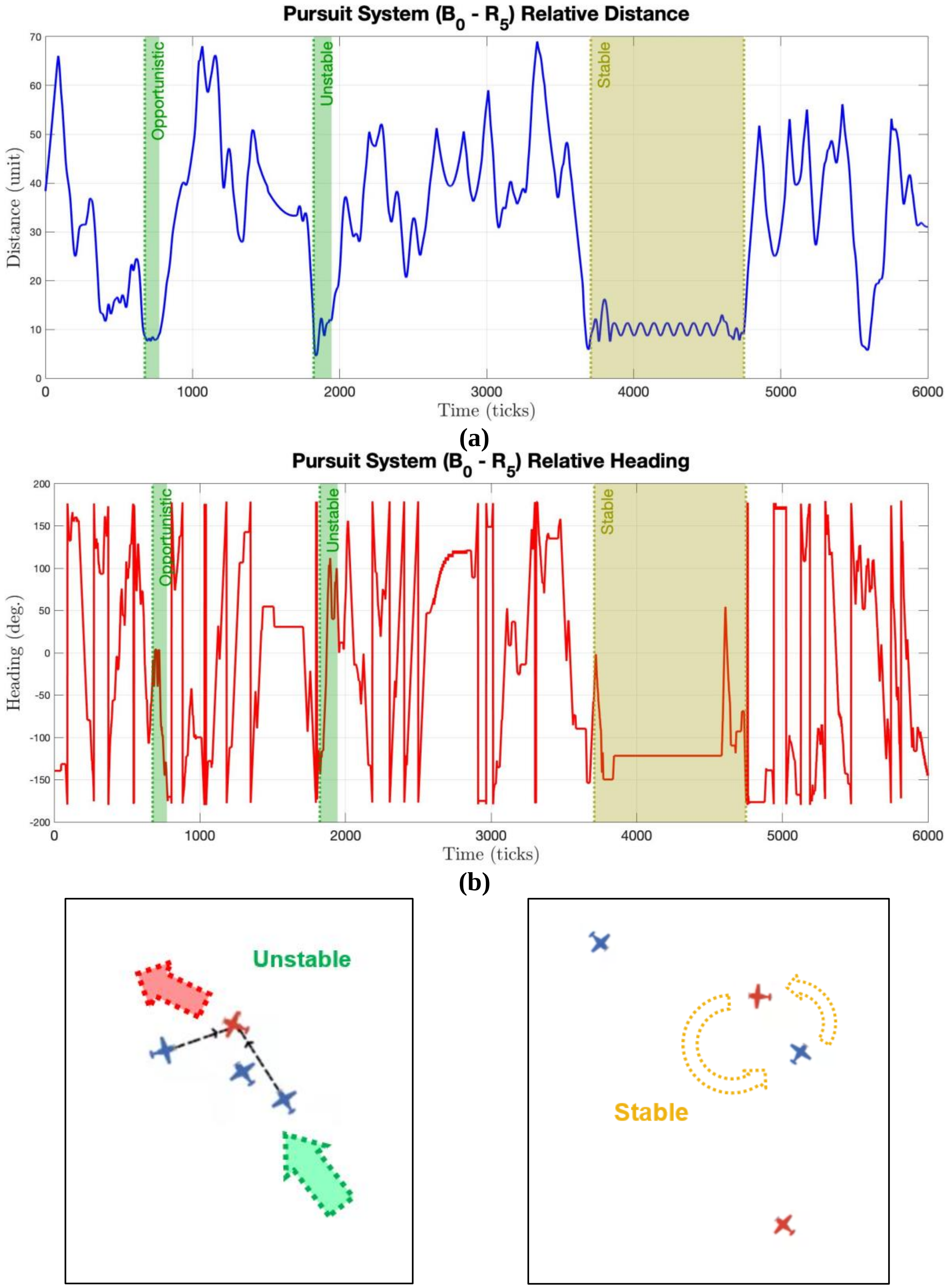


Figure 78 – Time series data corresponding to Adversarial Boids simulation (a) rel. distance between blue pilot 0 and red pilot 5, (b) relative heading, (c,d) screenshots

Experienced real-world pilots are unlikely to settle into a long-term pattern, but sporadic, short-lived instances of this self-organized pattern are likely. Unlike the friendly-pilot systems, the unstable systems depicted in Figure 78 vary in their structure. The system labeled “Unstable” is depicted in Figure 78c illustrates the case of multiple pilots chasing a maneuvering target. This formation appears to be unstable largely because each time one pilot engages, the others typically have to maneuver in order to avoid a collision, which introduces perturbations to their pair-wise distance curve. One way to account for these ternary and quaternary systems, is to have the interaction detection code generate and maintain a graph at each time step, where nodes are pilots and weighted edges represent interactions (0/1 for non/interacting pairs). It would be fairly straightforward to identify larger-scale systems by looking for clusters using the adjacency matrix.

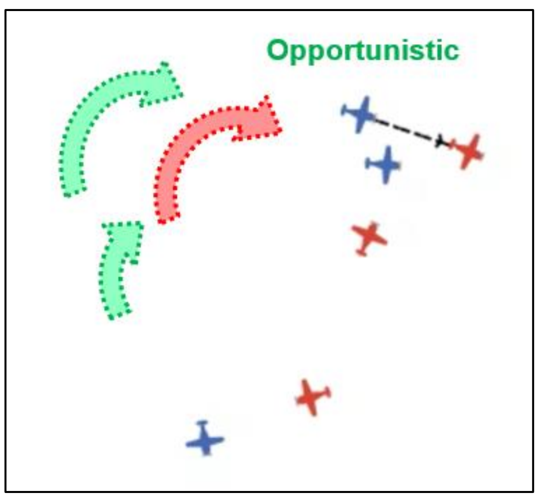


Figure 79 – Opportunistic attack indicated in Figure 78

Another unstable system is labeled “Opportunistic” in Figure 78 and is depicted in Figure 79. Here, one blue pilot shepherded a red pilot into the firing cone of another blue pilot. The green arrows indicate the trajectories of the blue pilots, and the red arrow indicates the trajectory of the red pilot. This system has a relative heading that briefly holds around zero (i.e. an approximately flat line) until, finally, the red pilot maneuvers out of the trap and forces the blue attacker to maneuver after it. These results show that the tools described in Section 5.2 are simple and effective ways to build a self-organization detection method. Furthermore, the extensions of this approach for more sophisticated self-organized objects are fairly straightforward (although they may be computationally expensive).

7.2 Sensitivity Analysis

Given that the distance between the red and blue pilots is simply the L-2 norm of the difference in their positions,

$$L_{pursuit} = \left\|\vec{P}_{blue} - \vec{P}_{red}\right\| \tag{64}$$

the rate of change of that distance can be found simply by taking the derivative,

$$\frac{dL_{pursuit}}{dt} = \frac{\vec{P}_{blue} - \vec{P}_{red}}{\left\|\vec{P}_{blue} - \vec{P}_{red}\right\|} \cdot \left(\frac{d\vec{P}_{blue}}{dt} - \frac{d\vec{P}_{red}}{dt}\right) \tag{65}$$

where the $(\cdot)$ is the dot product. It goes without saying that the goal of the pilot under attack is to increase the distance between it and its pursuer over time. From Eq. (65) it is clear that there are only two ways to achieve this goal. First, blue team can affect the blue pilot's velocity by improving the performance of the aircraft. Second, since the blue team cannot control the design of the red team aircraft, the red team pilot must be compelled to break off its pursuit somehow (thereby changing its velocity). The first option is a design change (means), while the second is a behavior change (ways). Although this approach does not specify exactly how either change can or should be achieved, it clearly outlines the options available to the decision-maker, and provides the basis for setting requirements for emergent behavior exploitation. It also associates those requirements with the components needed to achieve them through the upward causation equation, meaning that the emergent behavior interaction equation is not always needed. The equations of the self-organized system provide enough information for this approach in this case.

7.3 Hypothesis 3 Statement

In order for the length of the pursuit system to qualify as an emergent behavior, it must serve a function that another entity not contained within the pursuit system can act on. Drawing inspiration from the Thach Weave, one way to act on the separation between the bait and its attacker is to have the bait report that information to the hook and add a rule so that the hook responds to this information. There are several ways for that information to be reported. Some require appealing to the properties of the self-organized system, and others do not, which leads to Hypothesis 3:

> **<u>Hypothesis 3:</u>** *Targeting the system-level property will be more effective than targeting either pilot.*

In other words, this author suspects that when the bait communicates information about the pursuit system (as opposed to its own location, or the exact location of the attacker), the resulting response will be more effective. In order to measure that effectiveness, the MoMs must first be defined:

$$MoM_1 = \frac{Total\ Blue\ Team\ Shots\ Fired}{Total\ Red\ Team\ Shots\ Fired} \tag{66}$$

$$MoM_2 = \frac{Total\ Blue\ Team\ Shots\ Landed}{Total\ Red\ Team\ Shots\ Landed} \quad (67)$$

The ratio given by Eq. (66) describes blue team's ability to gain opportunities to attack red team, while the ratio given by Eq. (67) describes the effectiveness of blue team's attacks relative to red team's attacks. Aside from intuition, it is not at all clear exactly how the separation between pilots, L, from Eq. (64), will map to either MoM (i.e. the function that relates MoM to L is not obvious). Rather than attempt to identify that equation, this thesis will measure the values from experiment and infer how the various simulation settings affect L, and subsequently the MoMs.

Since the simulation environment contains all the information at once, one might wonder why the bait pilot should not explicitly report the position of the pursuit system (the center of gravity between the bait and the attacker). The main reason is that there is no meaningful causal relationship between the MoMs and the position of the pursuit system over the set of all simulations. Any single-variable function relating the two would be spurious, and a multi-variable function relating the two would be suspect. Having the blue pilot report the spacing between itself and its attacker is meaningful in that it can be associated with the MoM since the probability of firing and landing a shot are both functions of the distance between the attacker and target.

In a broader sense, testing Hypothesis 3 is important to this thesis for multiple reasons. First, empirical data showing that system-level equations are reliable, and useful cause-and-effect constructs undermines some of the more extreme reductionist arguments in philosophy, and places systems science on a stronger footing. Second, it must be demonstrated that the steps outlined in the method of this thesis provide useful information that has practical consequences on design and decision-making, and specifically on emergent behavior exploitation (in response to Research Question 3). The only way to show that the techniques in this thesis are effective practical tools is by experiment. The steps to test Hypothesis 3 in general will be outlined below.

1. For a given system-level behavior, perform the exploitation analysis step (sensitivity analysis).
2. Select one or more mission-appropriate MoM(s) and measure the performance of the system subject to the MoM(s) when it operates without the benefit of exploitation analysis information (no design changes or rule changes).
3. Implement the exploit identified in Step 1, and measure the impact of the exploit using the MoM.
4. Additionally, implement an analogous rule where the behavior of the component is modified to target a component-level property rather than the system-level property, and measure the subsequence change in MoM resulting from the change.
5. Compare the MoM values obtained for the exploit case to the baseline case (no modification whatsoever), to ensure that an adequate control case has been established.
6. Compare the MoM values obtained for the system-level rule change to the MoM values obtained for the component-level rule change. If the MoM for the component-level is superior to the MoM results for the system-level, then Hypothesis 3 has been falsified.

Since each scenario is different, the application of the exploitation analysis technique will result in unique considerations for the case studies in this chapter and CHAPTER 8.

7.4 H3 Falsification Test

To test Hypothesis 3 it will be necessary to compare situations where the pilots communicate system-level properties to situations where the pilots do not communicate system-level properties. This will be achieved by generating three sets of simulation data,: (C1) the bait pilot communicates its location to its wingman, (C2) the bait pilot communicates a position roughly corresponding to the attacker's position, in terms of the pursuit system length, and (C3) the bait pilot communicates a position roughly equal to position of the pursuit system. C3 will be treated as the "pursuit system property" case.

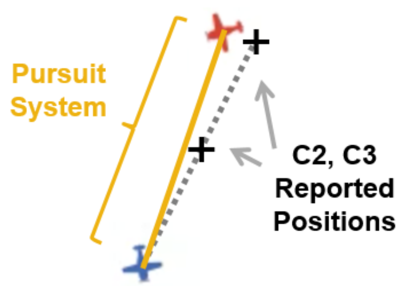


Figure 80 – Pursuit system (line) versus locations reported by bait pilot

As depicted in Figure 80, Cases 2 and 3 (C2-C3) may not always coincide with the exact physical location of the incoming attacker or the exact physical location of the pursuit system (i.e. the center of gravity between the attacker and bait pilots) because, in both cases, the bait pilot is communicating the pursuit system length and relative heading to the attacker *based on its latest observation*.[318] Due to NetLogo's randomized order of

[318] Analogous to saying "1,000 yards on my 5 o'clock"

execution, it is possible for the attacker to maneuver away from the position being reported during the same iteration that the bait pilot reports that information. Nevertheless, the errors in C2 and C3 are negligible for two reasons. First, they are numerically small to begin with. Second, the pilots have ample time to correct their headings because the distance over which the communications take place are at least one order of magnitude larger than the error itself.

For Hypothesis 3 to be supported, the MoMs obtained in Case 2 must be greater than the MoMs obtained in Case 1 and Case 3. The converse falsifies Hypothesis 3. This is the expected result because "reporting the pursuit system location" is roughly equivalent to having the wingman "lead the target." However, the reader is cautioned against reading too much into the idea of leading the target. That concept became common sense in real life due to mankind's interactions with nature. The simulation has no built-in sense of physics,[319] and the pilots have no built-in understanding of the projectile motion equations. Finally, note that the design change case does not contribute to testing Hypothesis 3. It is informative, however, to examine whether the design changes have a more significant impact on the MoMs than the behavior changes because that is the essence of a CBA.

7.4.1 Pilot Behavior Rules Modification

In order to perform a strict test of Hypothesis 3, the behavior modification must be implemented with the fewest changes possible. Attempting to explicitly implement a true Thach Weave would invalidate the experiment, because it would require multiple changes to the decision logic, relative to the design-change case. In this way, the design-change

[319] Although it can be extended for that purpose [380].

case serves as the "default behavior" control case. The least intrusive way to do this is to introduce an additional team of pilots called "Ghosts," depicted in Figure 81. During a blue pilot's observation step, the pilot stores the identifier of every enemy pilot within its vision cone. During the orient step, that pilot then determines if any of those enemies is maneuvering to attack (or had attacked during the previous iteration). The blue pilot now takes on the role of a bait pilot, and selects one enemy pilot as its biggest threat.

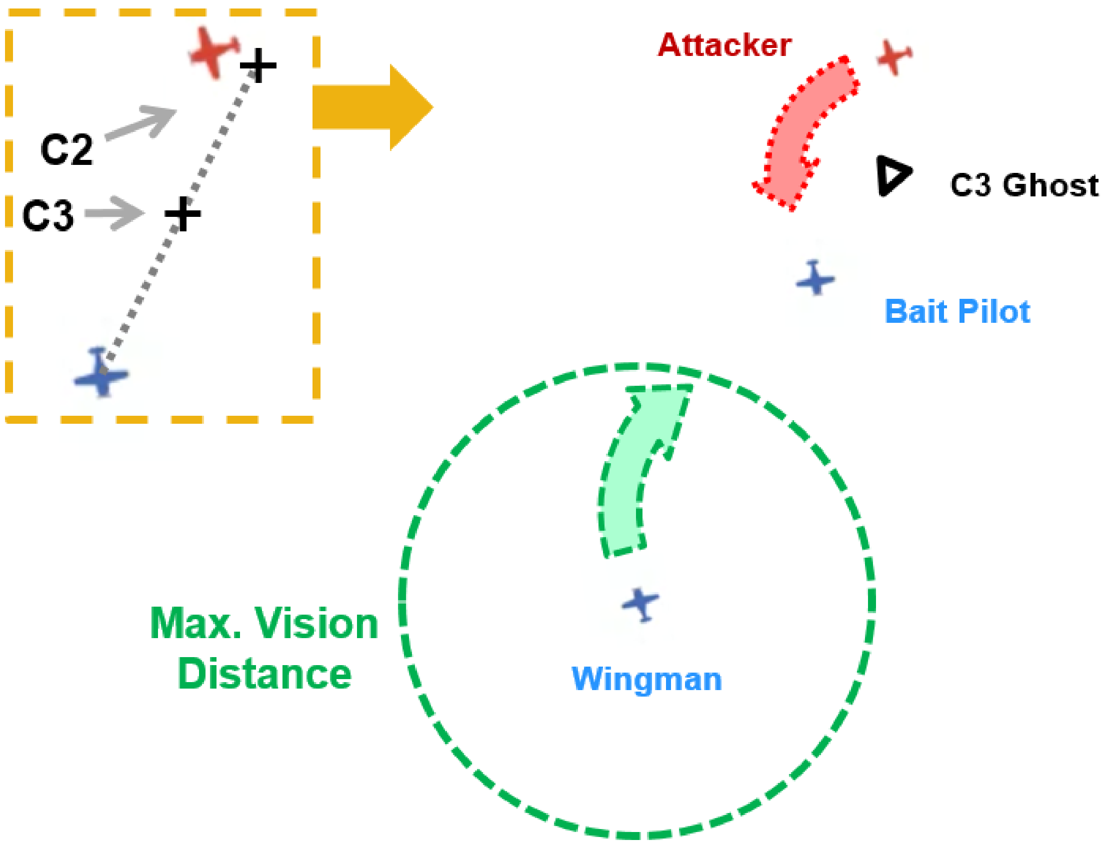


Figure 81 – Wingman turning to pursue ghost (black triangle) located approx. at pursuit system position (C3), outside of its max. vision distance

The pilot that fired during the previous iteration has priority in the threat ranking / decision curves. Upon selecting that threat, the bait pilot will initiate the creation of a ghost pilot in the simulation. The location of that ghost depends on the Case being run. Once the threat identification part of the orient step is completed, all pilots move on to the “assess reward” phase of the orient step, which includes determining if any visible enemies are viable targets, and determining if any visible teammates are in a position to form up with. During this step, the blue pilots will scan the map region outside of their vision cone for ghosts, and add them to their list of possible targets. Within their vision cones, the decision logic remains the same. Due to the behavior rules currently in place, the pilots will only act on the presence of a ghost if their vision cone is clear of enemies. Otherwise, they will simply pursue the most convenient target inside their vision cone. The action of detecting ghosts is equivalent to receiving a communication from another pilot about their attacker, but does not require any additional behavior rules. Assuming the pilot is not already engaged, and it detects a ghost, it will maneuver towards the nearest ghost, thereby dramatically increasing the likelihood of intercepting the attacker (relative to the design-change case, where the act of intercepting an attacker is purely coincidental). Ghosts inside a pilot’s vision cone are ignored. Finally, the ghosts are deleted at the end of each iteration.

This function, as implemented, acts solely as a mild override of the target assessment logic during the orient step (it simply increases the list of possible targets). No new decision curves are needed, and all other behavior remains the same. There are a number of obvious consequences to this mild change. First, this approach avoids problems caused by two bait pilots communicating to the same hook, which would require additional decision logic to resolve. Second, the wingman can get “distracted” by a new target that enters its vision field. Third, pilots may misidentify an incoming attacker (the pilots identify attackers based on their apparent behavior and proximity, and not in an omniscient sense). Fourth, a pilot only makes one ghost, per iteration. If that pilot is pursued by multiple attackers, the location of the ghost can jump around each iteration. Fifth, the only major difference between the behavior change cases and the design change cases is that the wingman simply turns around to rescue the bait if it isn't already preoccupied with another enemy pilot (in design change cases, it simply flies in a straight line until encountering another pilot in its vision cone). Once the hook enters within visual range of an enemy attacker, it will resume normal decision-making procedures. Therefore, sixthly, if there are multiple enemies in its vision cone by the time it arrives near the bait, the hook may decide to pursue an easier target rather than intercept the bait's attacker. Seventh, the ability to "detect a ghost from a distance" may or may not cause the pilots to break formation as they fly towards the bait. Eighth, with this communication ability, there is now a small benefit to flying in formation. Although the bait and hook do not have the full set of maneuvering instructions needed to consistently reproduce the Thach Weave, or even to report incoming threats to a wingman as in [185], the statistical significance of this communication is expected to be non-negligible, and to produce scenarios that mimic the Thach Weave.

7.4.2 Experiment Results

All of the data presented in this subsection is based on a sample of 1,500 simulations runs per case (5 cases in total), where each run lasts 6,000 iterations each. As shown by the box plots in Figure 82 - Figure 83, when the behavior rules are the same ($W_{R=B}$[320]), and both teams have the same design ($M_{R=B}$[321]), their performance is barely distinguishable. When red team is given a design advantage, its performance relative to blue team increases, as expected. However, the behavior change given to blue team

[320] As in, the ways (W) are the same (=) for red team (R) and blue team (B).
[321] The means (M).

substantially dwarfs the design advantage possessed by red team. The extent to which this change affects performance nonlinearly can be seen by the increase in the variance of the distributions, and the abundance of outliers. The rule changes do not guarantee that blue team will win every battle, but significantly improve its likelihood of doing so.

Since the ratios are difficult to see in these box plots, the MoM values are summarized via the bar plots Figure 84 - Figure 85. When the red team has a design advantage, and their rules are the same, the average number of shots fired and landed by blue team is substantially lower than those fired by red team: $MoM_1 \approx 0.7649$, while $MoM_2 \approx 0.7575$. However, when the means and ways are the same, both teams average the same number of opportunities to fire upon one another: $MoM_1 \approx 1.0036$, $MoM_2 \approx 1.0039$.[322] In other words, a 5% increase in maximum speed and maximum turn angle results in a nearly 25% increase in relative firing opportunities and lethality.[323] The ratios of shots fired for Cases 1 (W_{bait}), 2 (W_L), and 3 ($W_{L/2}$) respectively, are $MoM_1 \approx 1.491$, $MoM_1 \approx 1.4239$, and $MoM_1 \approx 1.5145$, while the ratios of shots landed are $MoM_1 \approx 1.4921$, $MoM_1 \approx 1.4282$, and $MoM_1 \approx 1.5226$. All behavior changes cause the blue team to obtain a roughly 50% improvement in its opportunities to fire (and land shots) on red team, despite red team's design advantage. However, of the three behavior changes, the rule to maneuver towards the pursuit system position provides the greatest benefit, which supports Hypothesis 3.

[322] This is consistent with the anticipated exact value $MoM_1 = MoM_2 = 1$, which somewhat verifies the model. Note that these MoMs were observed to converge to 1 as the number of simulations increases.
[323] Lethality roughly corresponds to MoM_2.

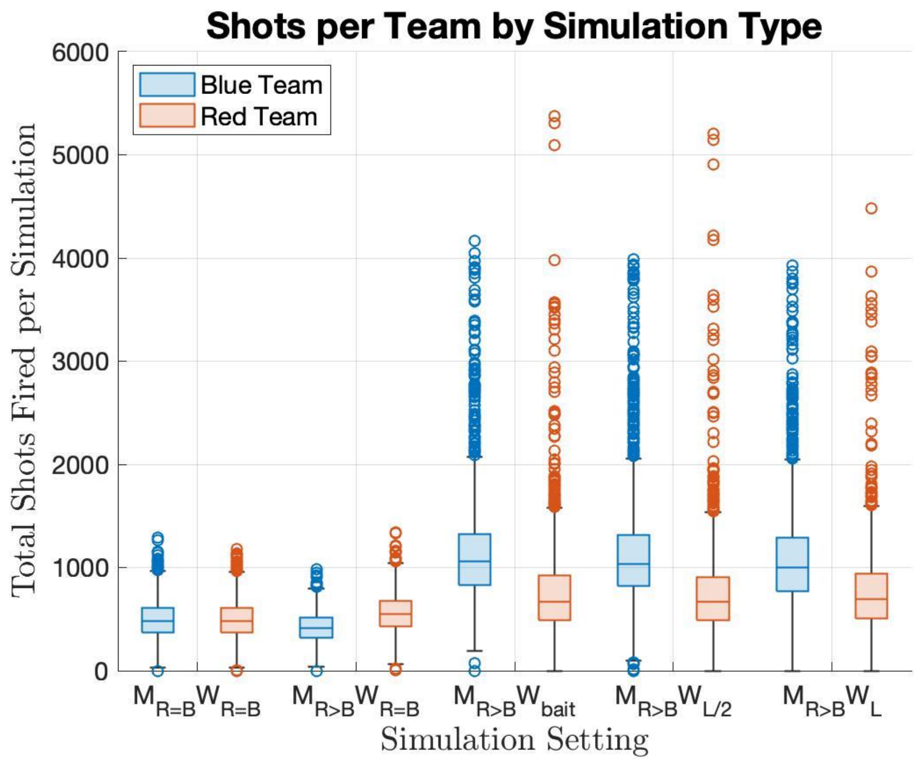


Figure 82 – Box plot of shots fired per team by simulation type

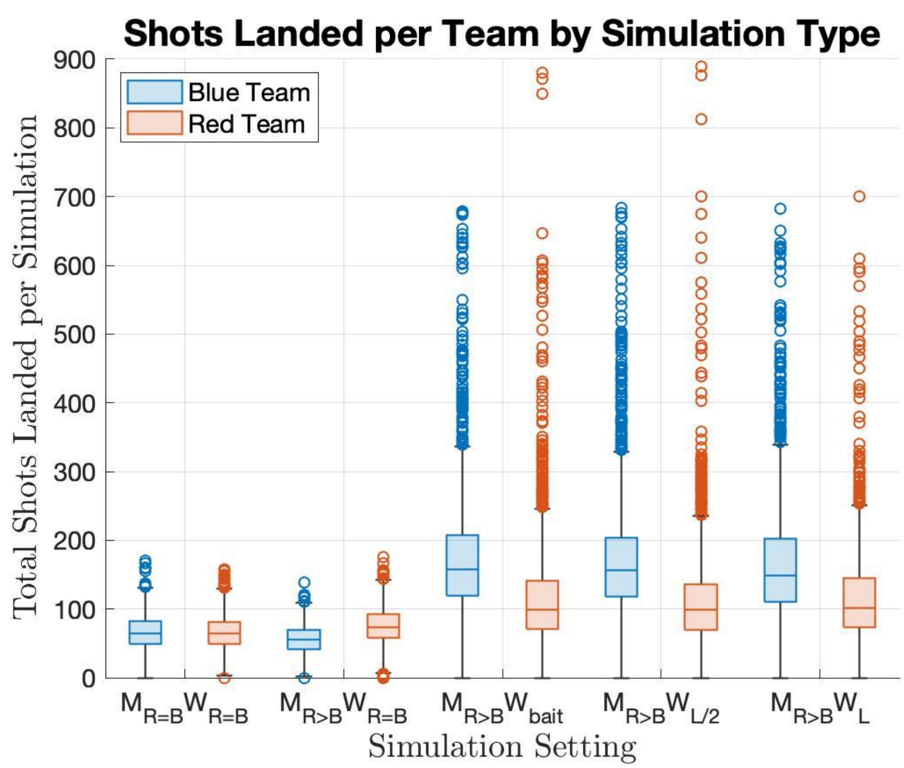


Figure 83 – Box plot of shots landed per team by simulation type

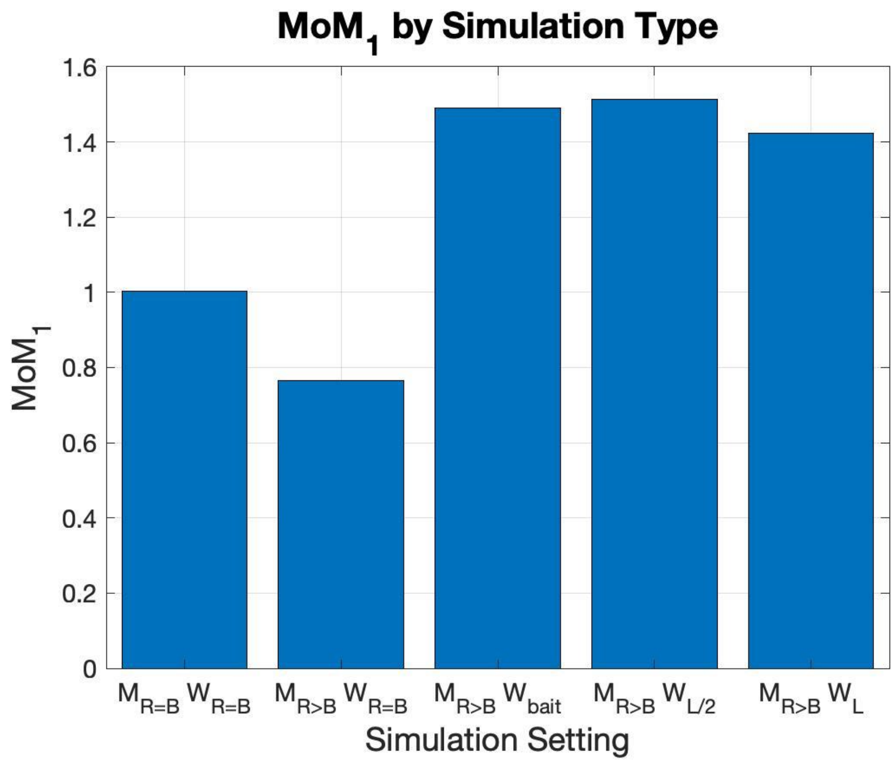


Figure 84 – Bar graph of MoM_1 by simulation

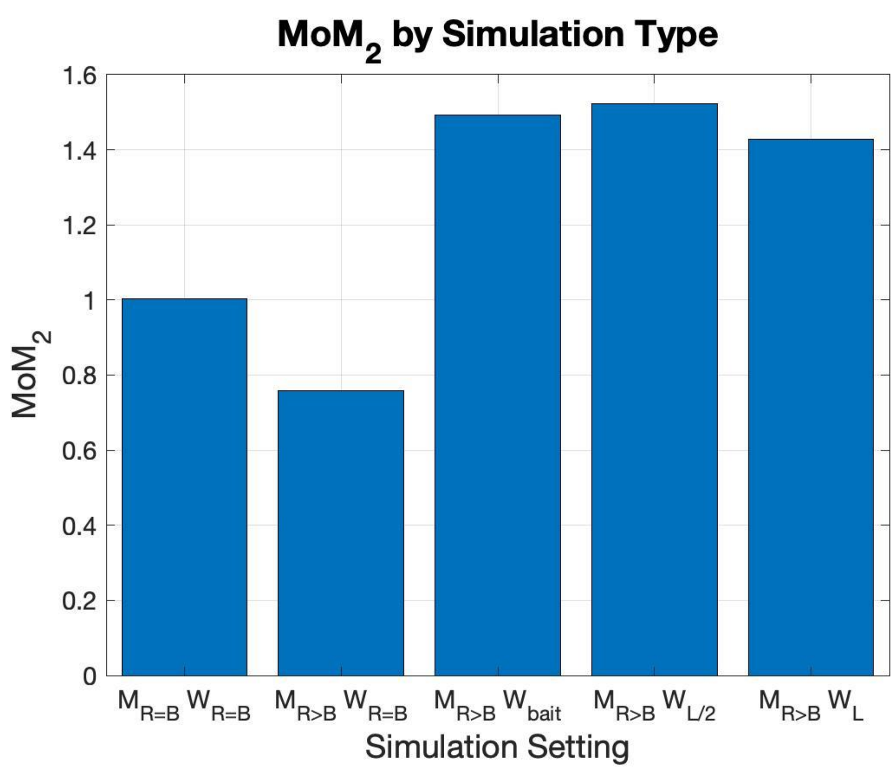


Figure 85 – Bar graph of MoM_2 by simulation

As seen in Figure 86b, when the two teams have the same design and rule set, the distribution of shots fired versus distance are the same. Neither team has a tactical advantage, therefore, the odds that pilots from either team will fire at any given distance are the same. Once red team is given a design advantage, it is able to fire more often from a closer range. Thus, the design advantage not only means that red team pilots can obtain firing opportunities more often, but also that they can obtain *favorable* firing positions (closer and aft) more often, which directly increases their lethality (as reflected in MoM_2).

The behavior rule change for blue team statistically dominates the design advantages available to red team, and fundamentally changes the nature of the dogfighting observed in the simulation. As indicated by the histograms in Figure 86c-e, the engagements for both teams tend to occur more often over a smaller range of distances than in Figure 86a-b. To put these distances into perspective, note that the maximum firing distance is 14 units, the "too-close" distance is 8 units, at which point pilots being weighing the risk of collision against other threats, and the distance at which pilots stop accelerating towards their targets is 10.4 units, at which point the pilots may choose to maintain or reduce their speed depending on the other behavior rules. The pilots have a clear preference for engagements in the range of 10-11 units, while engagements below this distance occur largely due to the pilot's momentum. No matter the distance, however, blue team always finds more firing opportunities than red team, which is precisely why the Thach Weave was developed, and supports Hypothesis 3.

Hypothesis 3 is supported *because the best values for both measures of merit occur when the blue team pilots are directed to fly towards the center of the pursuit system rather than either the bait pilot or the attacker.*

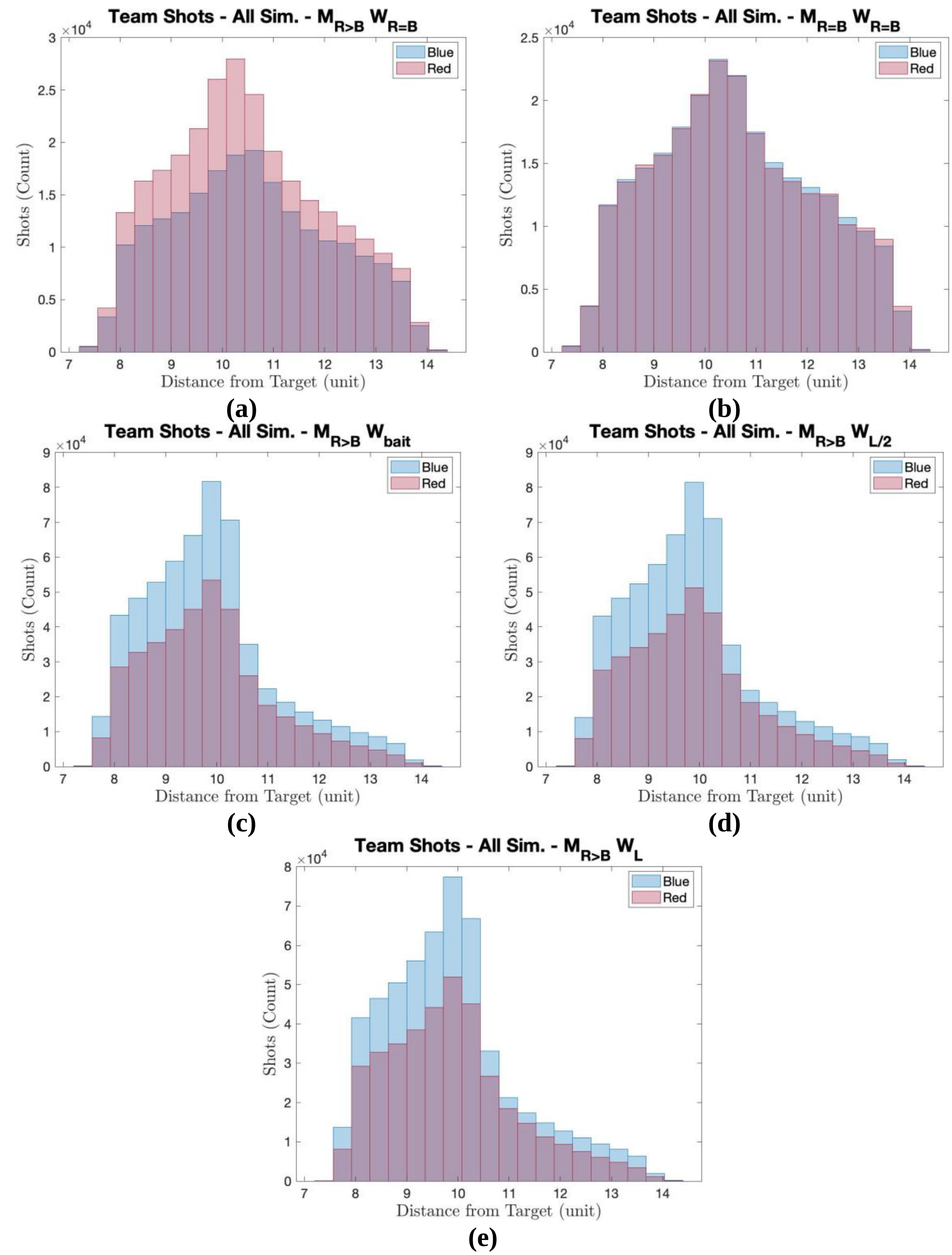


Figure 86 – Histogram of shots fired by Red/Blue teams over all simulations (a) $M_{R>B}$ $W_{R=B}$, red design superior (b) $M_{R=B}$ $W_{R=B}$, same designs/rules (d) $M_{R>B}$ W_{bait}, blue follow bait (e) $M_{R>B}$ $W_{L/2}$, blue follow system (f) $M_{R>B}$ W_{L}, blue follow attacker

7.4.3 Reductionist vs. Non-Reductionist Explanations

A typical reductionist explanation for the results in Section 7.4.2, mentioned briefly at the beginning of Section 7.4, might be that the behavior change works because the pilot is leading the target. Overall, the evidence does not support this interpretation because the performance substantially increased even when the pilot was aiming for the attacker. Thus, one might consider either of the following explanations to be a better alternative: (1) the change works because the communication causes pilots that would otherwise leave the scene to return immediately resulting in blue team's pilots being outnumbered less often than red team's, or (2) the change works because the simulation predictably generates pursuit systems that have a finite length, and the team that capitalizes on the location of the pursuit system will have a tactical advantage over a team that cannot. Both alternative explanations are valid, and both are expressed in terms of system-level properties. The first explanation implicitly relies on the ratio of combatants that form a multi-pilot pursuit system, while the second explicitly references the creation of a new self-organized system and its useful properties. Both system-level explanations can be used to write simple behavior rule modifications that measurably improve performance. The data generated for this test says nothing about whether the behavior rule "always lead the target" would enhance pilot performance, but given that the shots land instantaneously, leading the target would only serve as a rule for maneuvering, not firing. It is unclear that causing pilots to overshoot their targets early on in their attack run would improve their performance.

7.4.4 A Remark on Hypothesis Testing

Recall that, for blue-team, the *best-case* performance in firing opportunities in Figure 82 comes from maneuvering towards the bait (W_{bait}), the biggest improvement in *worst-case* performance comes from maneuvering towards the attacker, and the best *average* performance comes from maneuvering to the center of the pursuit system. On its face, the various statistics present a kind of conflict between the possible conclusions one could draw from the data. With regards to Hypothesis 3, it is clear that the term "effective" contains an inherent subjectivity, and so the user must be careful to avoid motivated thinking when interpreting results. A critical examination of Figure 82 shows that the median total shots fired of the blue team with behavior modification is consistently higher than the upper quartile of the red team's results. Thus, there is no pragmatic reason to falsify Hypothesis 3. Nevertheless, had a one relied on the best-case or worst-case statistics, one could have drawn a very different conclusion.

7.4.5 A Persistent Ambiguity

The case where L/2 is communicated brings back the question of whether the pursuit system's position should be treated as the emergent property (rather than length). The emergent *behavior* of the system would be to widen its length over time, but the information needed to achieve this goal is the position of the system. This possibility was rejected in Section 7.3 on account of the fact that neither MoM can be expressed as a meaningful function of the system position. Given that the position of the pursuit system requires two variables to describe, while length is only one, there seems to be a potential contradiction in the case where the same information can be represented two ways. For the purposes of hypothesis testing, this thesis selects the worst-case scenario (position). However, this has a direct impact on Hypothesis 1, and one's ability to calculate the amount of data compressed by self-organization, and should be considered in future studies.

Another ambiguity (one that affects the discussion in Section 7.4.3) is whether the "number of pilots" is a property of the system only, or a property of the pilots as well. In this author's personal experience, it is not uncommon for engineers to assign trivial properties to objects whenever some new information suggests the need for generalization. For example, one could argue that a boid is a flock where $L = 0$, and $n = 1$. This is not unprecedented. Tautologies and vacuous truths play an important role in the development of a logical framework. The same is true for trivial statements. However, with regards to SE, this presents an opportunity for category fallacies that can undermine the rigor of decomposition techniques. This thesis treats the "number of elements" exclusively as a property of a set, and not a property of the element of a set. There is no need, at this time, to settle the question dogmatically, but this ambiguity is likely to make some ontologies more effective than others with regards to emergent behavior identification. This thesis leans towards driving a hard distinction, with the exception of systems that can be defined in a purely recursive manner. In SE, it necessary to make unambiguous statements such as: "a set is not an element," "a set can be an element within another set," and "an element can belong to multiple sets." It is also necessary to say that "the mass of a proton is distinct from the mass of a quark or the mass of a ship," just as it is important to say "the position of a ship is distinct from the position of a fleet." The same concept can be mathematically computed in different ways (as is the case with mass), or serve totally different and unrelated functions (as is the case with position).

CHAPTER 8. UAV SWARMING CASE STUDY

In this chapter, the complete method presented in CHAPTER 5 will be applied to a simulation of Unmanned Aerial Vehicles (UAVs) flying in a simulated city. The simulation environment, called SwarmLab, was developed by Soria, Schiano, and Floreano from the Laboratory of Intelligent Systems in Switzerland [254]. This experiment will utilize the environment's ability to simulate swarms of fixed-wing and quadcopter drones navigating obstacles according to a set of decentralized behavior rules. The environment offers two options for drone decision-making: (1) rules based on work by Vásárhelyi et al. (VEA) [255], and (2) rules based on work by Olfati-Saber and Murray (OSM) [256].

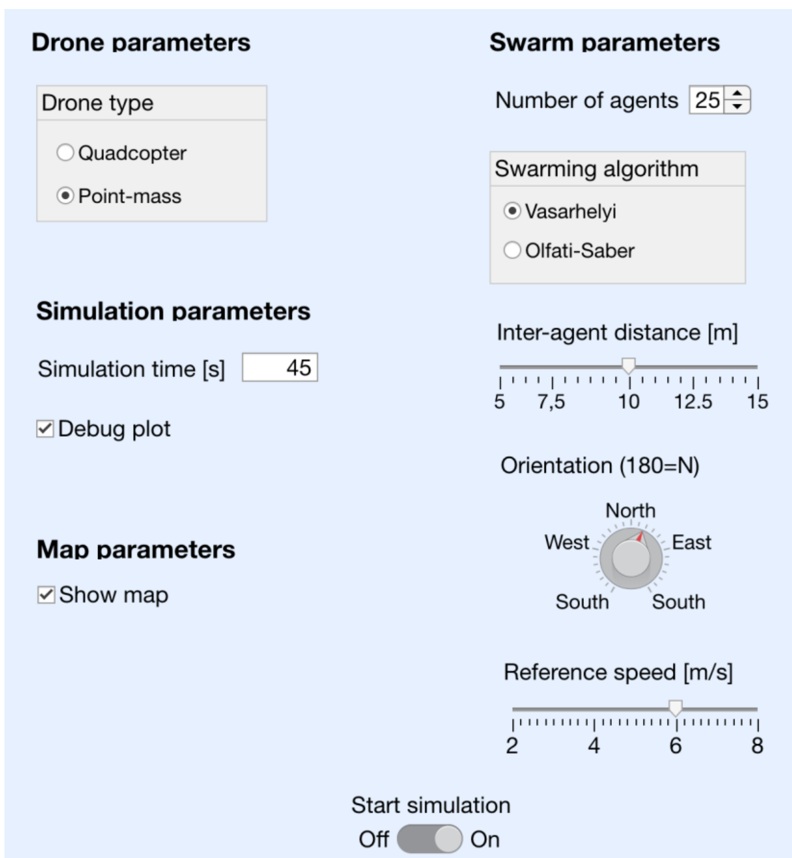


Figure 87 – Sample of SwarmLab GUI with notional settings

Figure 87 shows the options available via the SwarmLab GUI. These settings will be used in the discussion that follows (only the swarming algorithm / control law is varied). The performance metrics tracked by SwarmLab will be discussed in Section 8.3.1.

The OSM rules are designed to enable drones in a swarm to maneuver individually such that the swarm's collective shape is preserved to the greatest extent possible while moving through a space. To achieve this, Olfati-Saber defines a graph that, under certain conditions, corresponds to a unique and unambiguous[324] swarm formation. Using the desired properties of this graph, a so-called "structural potential function" is obtained, and the gradient of this potential is used as to derive a smooth, bounded, nonlinear state feedback law that ensures "collision-free local stabilization" of the drones in the swarm, again, under certain conditions [256].[325]

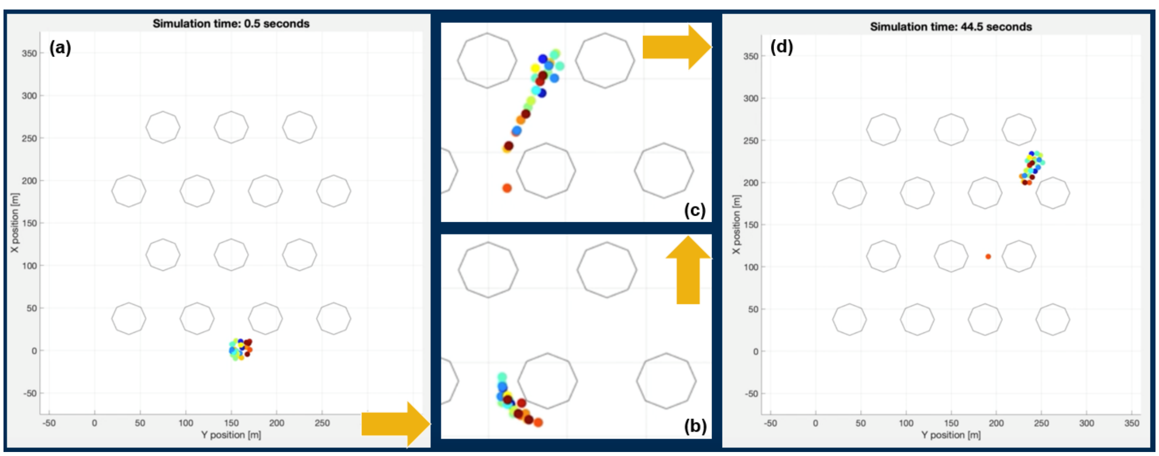


Figure 88 – SwarmLab plots of 25 point-mass drones navigating obstacles according to OSM control law at various time steps (a) 0.5s, (b) 6.5s, (c) 22s, (d) 44.5s

[324] The terms "unique" and "unambiguous" take on a specific mathematical meaning in [311]. However, the layman's definition of these terms is close enough for the purposes of this narrative.
[325] Again, each of these terms has a specific mathematical meaning. See [311] for details.

Figure 88 depicts four snapshots of a 25-drone simulation wherein the swarm attempts to move from the south side of a simulated city (bottom), to the northeast side (top-right corner). The drones are following an implementation of the OSM control laws, and the simulation runs for 45 seconds. Figure 88b depicts the swarm pressing into itself as it encounters a building obstructing its path. After the swarm slowly maneuvers west and north of the building, most it the drones are able to regroup, but ultimately one drone falls behind, seen in Figure 88c-d. Aside from the solitary drone, the remaining drones generally retain their lattice-like configuration, as intended by the control law.

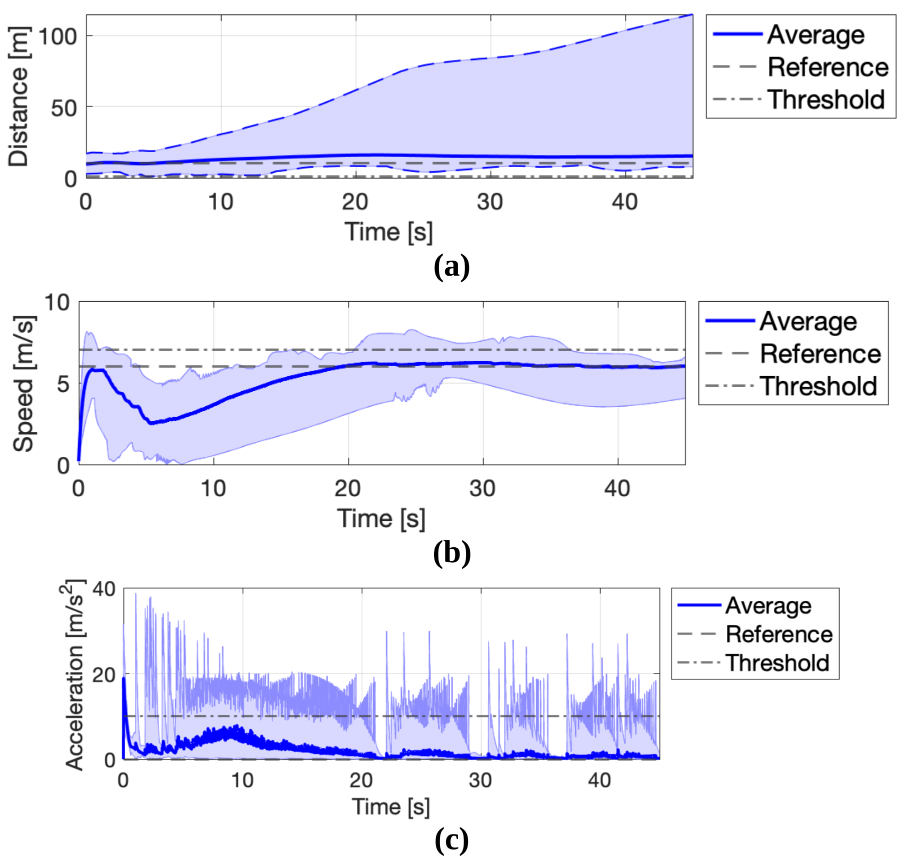


Figure 89 – SwarmLab time series plots of min/average/max properties for drones following OM rules (a) distance, (b) speed, and (c) acceleration

As seen in Figure 89a, the maximum inter-drone distance increases with time due to the single drone that fell behind. The average separation between drones remains fairly constant near the reference value, however, there is one instance before the 5s mark where the minimum separation between drones drops below the threshold, suggesting a collision may have occurred. The drone speeds (Figure 89b) and acceleration (Figure 89c) repeatedly exceed the desired threshold, suggesting potentially unsafe operation. Nevertheless, the swarm covers most of the distance required in the time allotted.

The VEA control law was designed in response to what the authors perceived as "reality gap." Rather than assuming an idealized shape, as in the OM case, or idealized environment with perfect communication between drones, the authors drew on an initial rule set derived by Reynolds [188], and then added more rules in order compensate for various real-world challenges. The VEA control law contains 11 tuning parameters, and used an evolutionary algorithm and 15,000 objective function evaluations to obtain robust values. The VEA control law was then tested using a real-world swarm of 30 quadcopters.

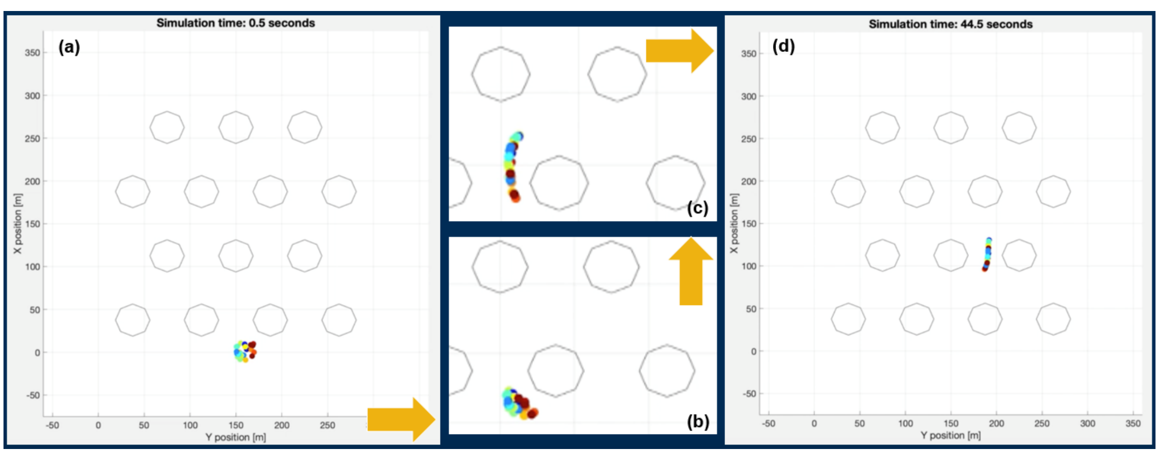


Figure 90 – SwarmLab plots of 25 point-mass drones navigating obstacles according to VEA control law at various time steps (a) 0.5s, (b) 6.5s, (c) 22s, (d) 44.5s

As shown in Figure 90, the swam obeying the VEA rules retains is cohesion better than the OM swarm, avoids obstacles as a unit rather than "smearing" across the surface of the obstruction, and maintains a greater distance from the obstructions. However, the swarm also moves slower, covering only about half the distance the OM swarm travelled.

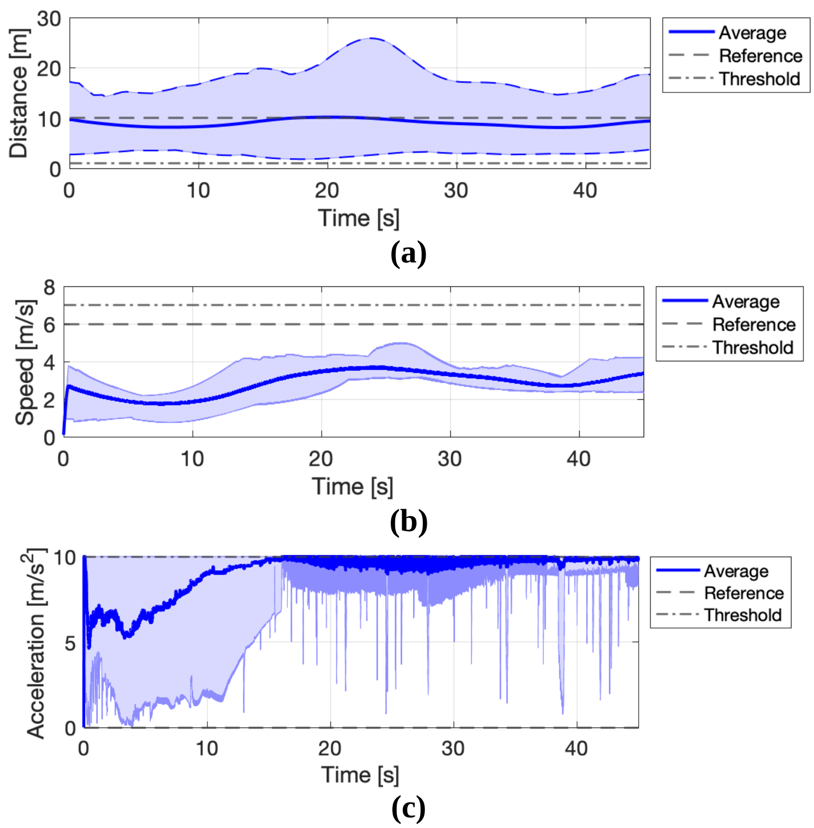


Figure 91 – SwarmLab plots time series plots of min/average/max properties for drones following VEA rules (a) distance, (b) speed, and (c) acceleration

The VEA swarm never exceeds the speed or acceleration thresholds, nor does it violate the distance threshold. However, unlike the OM swarm, it does not retain a lattice shape, and flattens out into a snake-like arrangement when traversing an environment with multiple nearby obstacles (see Figure 90c-d).

8.1 Pattern Recognition

This section will examine the behavior of an OM swarm flying three missions: (1) a simple patrol around a set of buildings, (2) traversing a small number of buildings directly obstructing the swarm, and (3) traversing a narrow passage between buildings. Unlike the (adversarial) boids cases, the drones are initially placed in close proximity. Therefore, this case will examine disruptions to the swarm shape more so than its formation.

8.1.1 Simulation Settings and Modifications

The dynamics of point-mass drones are very different from those of quadcopters simulated in SwarmLab. Quadcopter simulations take into consideration the behavior of the actuators on the quadcopter, as well as the dynamics and kinematics of the drone.

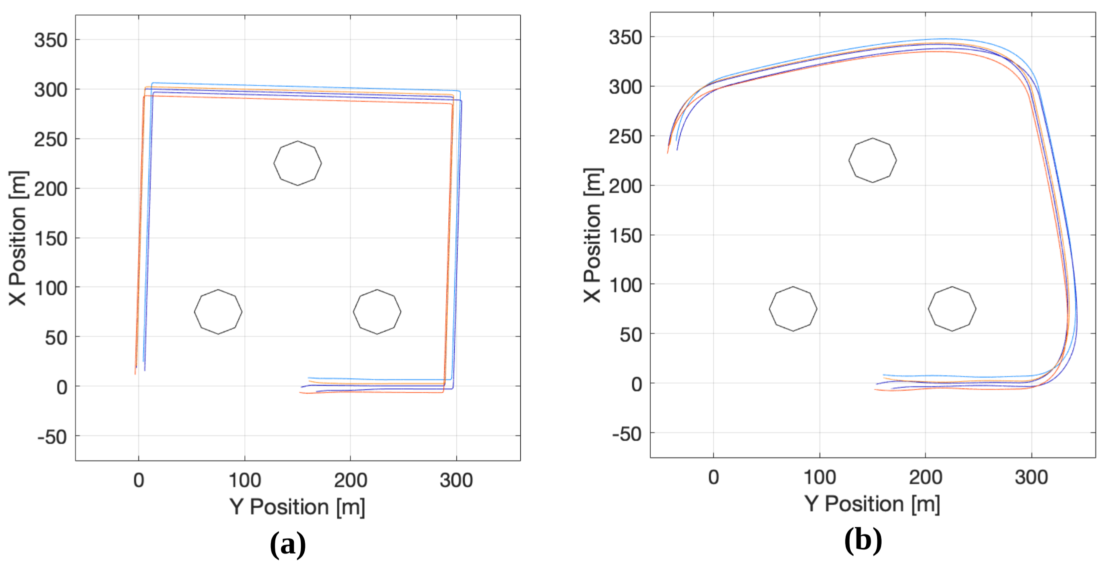


Figure 92 – Paths traced by 5-drone swarms visiting four points around a set of obstacles within 200 seconds (a) point-mass drones, (b) quadcopters

Point-mass drones, on the other hand, have their positions updated using a simple Euler forward method [254]. As seen in Figure 92b, quadcopter swarms will overshoot waypoints when tracing a path[326] around a set of buildings due to their inertia, whereas the point-mass drones perform much tighter turns (Figure 92a). Note, also, that the traces in Figure 92 are top-down views of 3-dimensional motion. That is, the drones can fly above/below each other, so long as they maintain a safe distance, and the swarm remains fairly level.

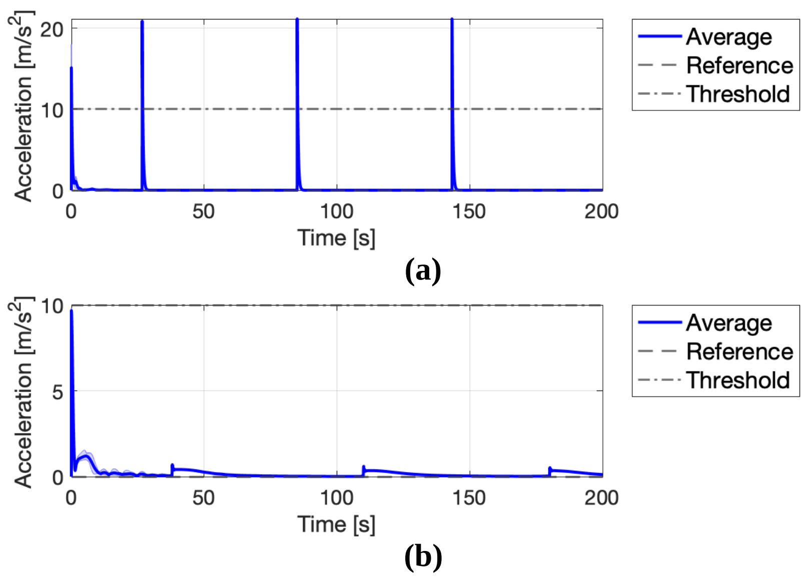
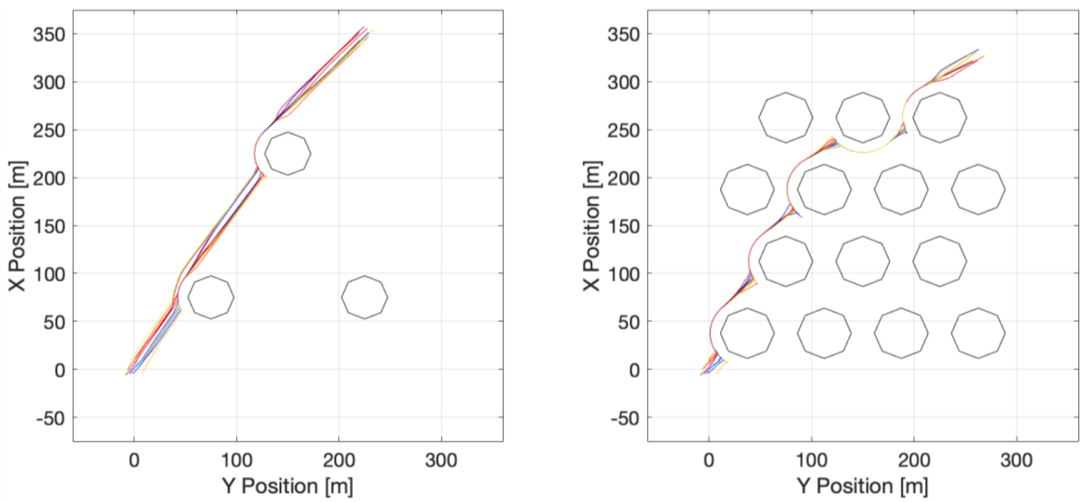


Figure 93 – Acceleration versus time plots for mission in Figure 93 (a) point-mass drones, (b) quadcopters

In order to achieve these maneuvers, the point-mass drones undergo unrealistic spikes in acceleration that exceed their thresholds (compare Figure 93a-b). Despite this, the point-mass drones do not exhibit any other pathological behavior (the acceleration is too brief to cause a collision). Therefore, all remaining simulations in this chapter will use point-mass drones and will neglect wind effects in order to reduce simulation run time. Note that the quadcopters in Figure 93b also exhibit an unrealistic initial acceleration. This numerical

[326] These drones are using the OM control law, and are simply directed to fly towards the points (0,300), (300,300), (300,0), and (0,0) in counter-clockwise order.

artifact is easily remedied by initializing the swarm with the velocity derived from its mission requirements rather than zero velocity.



(a) (b)

Figure 94 – Paths traced by 10-drone point-mass swarms (a) across two obstacles in 120 seconds, (b) through a narrow passage in 160 seconds

In the remaining missions (Figure 94), the swarms retain their cohesion while navigating obstacles. Cohesion, which depends partly on the spacing between obstacles and the number of drones in the swarm, is not always guaranteed, as seen in [254]. The paths available to the swarm in Figure 94b are just wide enough to be traversed without splitting into smaller clusters.

Unlike the boids case, this chapter will deal with self-organized objects that are largely amorphous and flexible in three dimensions. Unlike the (adversarial) boids cases, this section will deal with environments that contain obstacles, and consider the impact of those obstacles on the robustness of the pattern-recognition process. The SwarmLab simulation space is unbounded in the horizontal plane, and semi-infinite in the vertical direction. Finally, unlike NetLogo, the drones do not execute their instructions in random order each iteration. The code was extended so that the swarms do execute their instructions in random order, but, within each swarm step, the drones will execute their instructions in the order determined by their index number.

8.1.2 Results

First consider the separation between drones during each mission. In Figure 95, below, “PM” stands for point-mass and “Quad.” is simply the abbreviation for quadcopter.

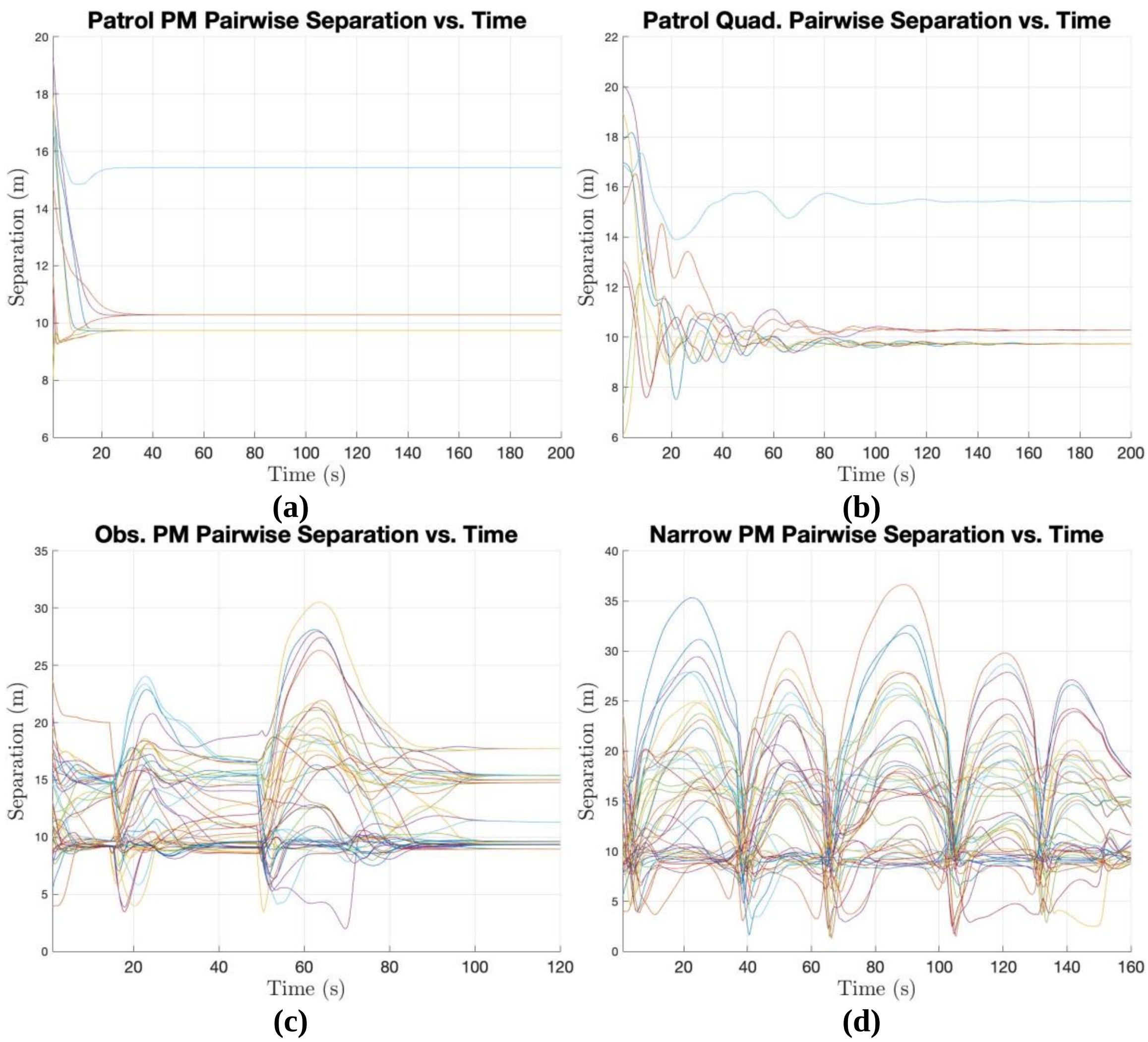


Figure 95 – Inter-drone separation during missions (a) patrol by point-mass (b) patrol by quadcopter, (c) obstacle traversal by point-mass, (d) narrow pass traversal by point-mass

The point-mass swarm exhibits stable behavior for the majority of the patrol mission (Figure 95a) due to the absence of obstacles and its ability to change translation directions without undergoing a rotation. The initial variation in separation is due to the random initial placement of drones. Although they are very close to one-another, they are not in an optimally stable formation according to the OM control law, and so they gradually find a stable arrangement as they travel to their first waypoint. The separations between drones in the stable 5-drone swarm are 9.7m, 10.3m, and 15.4m (10m is the user-defined "reference" separation). These values can be used to filter subsequent simulation data, as will be discussed towards the end of this section. The 5-drone quadcopter swarm flying the patrol mission (Figure 95b) takes much longer to stabilize from its random initialization, and the first turn it encounters during its mission (near the 60 second mark) de-stabilizes the swarm slightly. For the purposes of real-world self-organization detection, however, the quadcopter simulation suggests that a pattern recognition procedure would either require some tolerance for small perturbations, or perhaps the use of a moving average.[317] The 10-drone point-mass swarms (Figure 95c-d) show significant perturbations to their stable operation as they attempt to avoid obstacles. Upon reaching an obstacle, the swarm undergoes a contraction followed by an expansion, until the opportunity presents itself for the swarm to re-stabilize. However, as with the patrol mission, most drones are able to maintain a separation of less than 15m.

Figure 96 shows the corresponding pairwise, normalized velocity dot products for the various missions. The patrol missions show smooth, parallel flight throughout the mission for both point-mass and quadcopter swarms. Thus, the OM control law is able to maintain uniformity across drone velocities as they perform simple maneuvers.

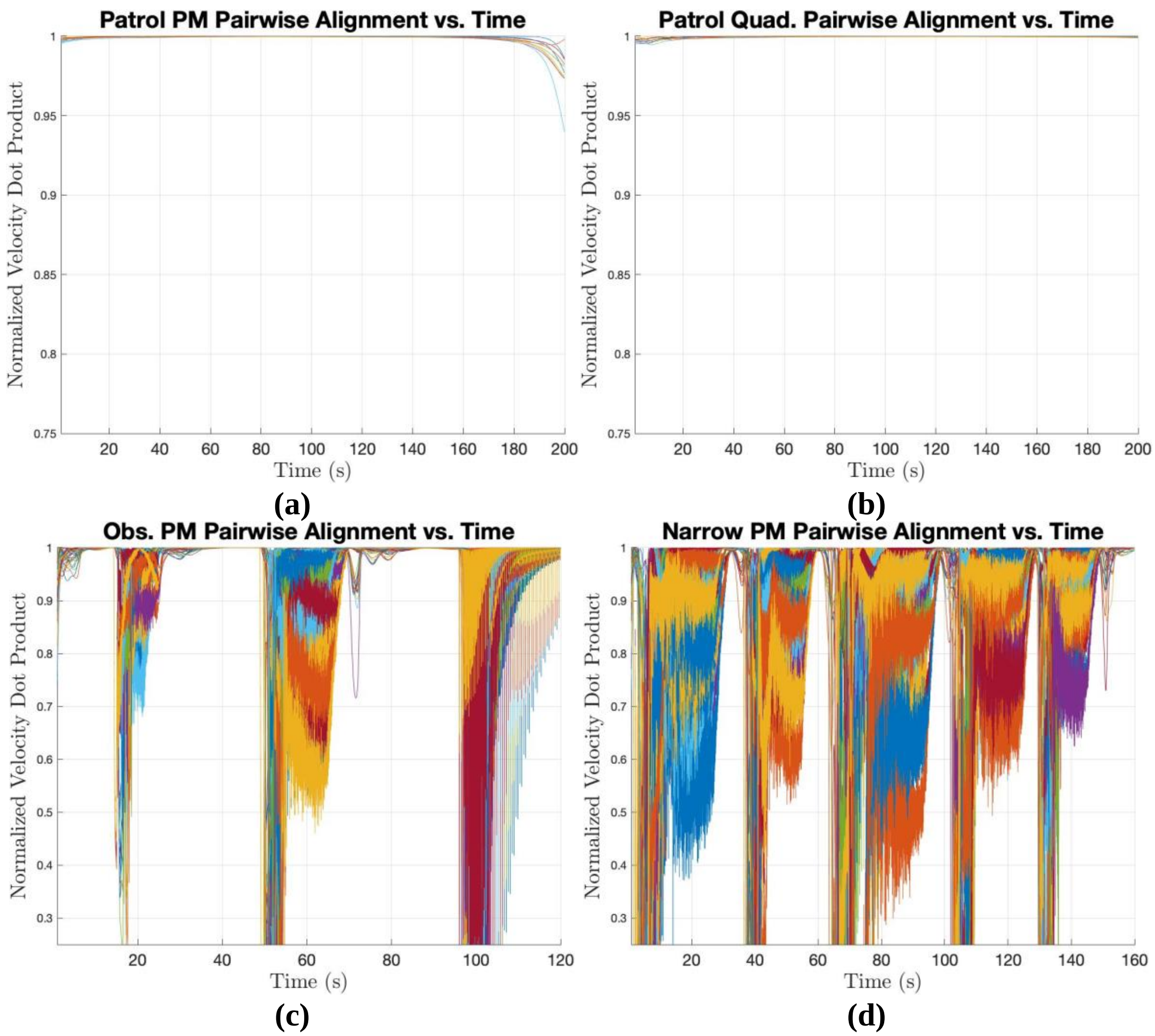
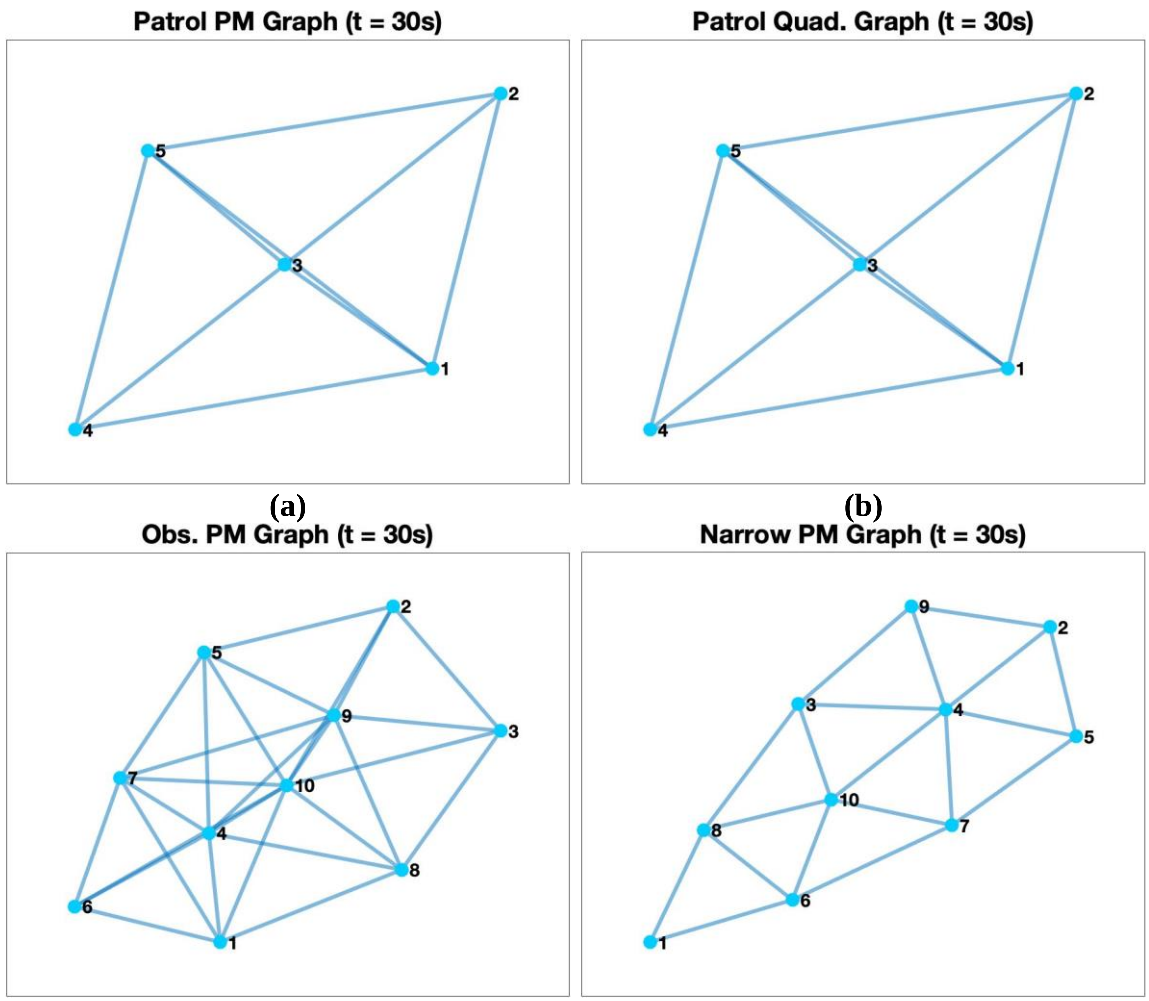


Figure 96 – Inter-drone alignment during missions (a) patrol by point-mass (b) patrol by quadcopter, (c) obstacle traversal by point-mass, (d) narrow pass traversal by point-mass

For the missions where the swarm must avoid buildings (Figure 96c-d) the alignment profiles are significantly more chaotic. This is undoubtedly due to the unrealistic accelerations of the point-mass drones. Nevertheless, the deviations clearly indicate periods of stable flight punctuated by significant disorder when the swarm encounters a building. The initial deviations are very small by comparison.

Since stable swarm motion is associated with the majority of drone distances being less than 15m, it is possible to use, say, 14m as a threshold for identifying stable "neighborhoods" for each drone (this feature is built in to SwarmLab).



(c) (d)

Figure 97 – Graphical representation of swarm adjacency matrix at 30s mark (a) patrol by point-mass (b) patrol by quadcopter, (c) obstacle traversal by point-mass, (d) narrow pass traversal by point-mass

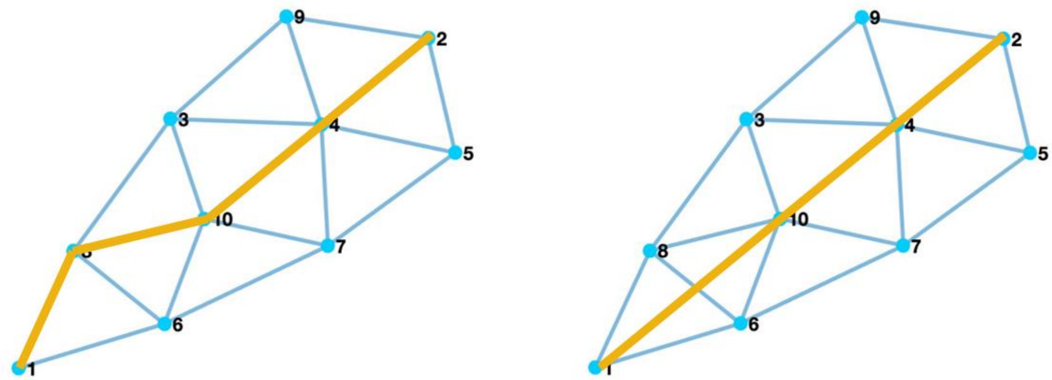
With that information, it then becomes possible to identify a graph for the swarm, along with its adjacency matrix, that correspond to (meta)stable swarms. This provides a sense of structure to what is otherwise an amorphous cloud of points, and one can check that the graph remains connected to determine whether a drone has broken from the group.[327] Graphs for the sixty-second mark during each mission are provided in Figure 97. In practice, one would take additional factors into consideration such as communication distance, line of sight, and any other sensors the drones can use to maintain contact with one-another.[328] This additional feature will serve as the "interaction detection" among drones, and can be extended to the adversarial case that will be discussed in Section 8.3.

8.2 Behavior Association

As with the (adversarial) boids models, this section will examine the swarm speed, and heading to determine if they are emergent behaviors. Since the drone movements are fully three-dimensional, the pitch and yaw of the swarm will be treated as its headings. Since the drones are a point-mass, and the swarm is easily distorted, the roll angle will be left for future study. There are at least two ways to compute a characteristic length for a swarm. The first can be obtained by computing the shortest path (based on the swarm's distance-weighted adjacency matrix computed using the aforementioned 14m cut-off distance) between the two drones whose Euclidean separation is the largest.[329] The second is the largest separation between any two drones along the swarm's principal axis (this will be referred to as the length of the swarm's principal axis). If the swarm were a perfect sphere, the first length equal half the circumference of the circle inscribed onto a plane

[327] An alternative would be k-means clustering, but if the number of clusters is not known a prior, finding them using k-means can be cumbersome.
[328] SwarmLab contains some of this functionality. However, since this thesis is focused on self-organized structures, it is more meaningful to define neighbors based on stability rather than communication range.
[329] In the unlikely case of a tie in Euclidean distances, the shortest path among the various alternatives will be selected.

passing through the swarm's center, and the second length would be the diameter of the swarm. Another example is provided in Figure 98.



(a) **(b)**

Figure 98 – Depictions (highlighted in yellow) of (a) the shortest path between the furthest points, and (b) line segment corresponding to largest separation along the principal axis of a 2-dim set of points

Properties of the swarm can also be derived from its control laws. This thesis will consider the average separation between drones as one such property.

Vasarhelyi et al. loosely extend the concept of mechanical pressure to swarms of drones. This concept was first discussed with respect to large crowds of people [257] [258], where individuals may push against one another, thereby exerting a literal force that can be averaged over the size of the crowd. Vasarhelyi et al. only use the term to facilitate explanations, and do not propose a means for computing this swarm pressure. Any attempt to extend the usual thermodynamic or kinematic definitions of pressure to drone swarms would require careful study and experimentation. True drone collisions are nothing like the simple elastic collisions typically assumed in physics, and the consequences of such collisions are incompatible with the models of pressure in physics. Collision avoidance maneuvers are much closer to the physics notion of an (in)elastic collision, but modeling them as such depends on the mechanics of the drone itself. That is, a different model would have to be created for each type of maneuver, including the number and types of drones involved in the collision. Such derivations are outside the scope of this thesis.

Finally, note that the communication range of each drone also affects the swarm's structure. The de-centralized control laws provided in [254] ensure some kind of local stability, which also ensures global stability under certain circumstances (depending on the assumption underlying the control law). In the case where the swarm size exceeds the communication range of the drones, the swarm will deform into shapes that are everywhere locally stable, but not necessarily globally stable. This thesis will consider the simplest case where all drones remain within each other's communication range. Note that SwarmLab computes neighborhoods based the drone communication range for the purposes of calculating performance metrics such as swarm safety. The code was extended to perform an additional adjacency matrix calculation based on the 14m cut-off distance.

8.2.1 Simulation Settings and Modifications

The simulation set up is very similar to the boids model. A series of 1,328 simulations[330] was run wherein two 10-drone swarms are set on a collision course in an obstacle-free environment. The initial drone distributions were random, and so the swarms were spaced far enough apart that each one could stabilize before encountering the

[330] The original goal was to run 1,500 simulations. The number of simulations was cut for time, but as of this writing, a more efficient way of running the code has been found (i.e. the inefficiency was not due to SwarmLab). Nevertheless, since the control laws enable the swarms to remain fairly stable, 900 of the 1,328 simulations resulted in unbroken swarms. Compare that to 313 unbroken flocks out of 5,000 Boids model simulations run for Section 6.2.

opposing swarm. Unlike stable flocks of boids, the drones rarely fly at perfectly constant velocities, but the variations are small enough that stable flight can be easily identified: the magnitude of the maximum acceleration for any drone would drop below 0.5 m/s^2 (this same criteria was used to help determine when the swarms had finally re-stabilized). The swarms were given trajectories that intersect at a variety of (yaw) angles, beginning with 180°, which is a head-on collision, and decreasing to 90° by increments of 5°. These initial trajectories, as well as the swarm initial positions, are then perturbed randomly to produce the full 1,328 collision simulations. The swarm positions were perturbed by no more than ±3m in either horizontal direction (all swarms were initialized at the same elevation). The swarm trajectories were perturbed by no more than ±4°. Therefore, some collisions were glancing collisions while others were direct. Unlike the boids model, however, the initial trajectories of both swarms were varied so that no swarm started at the same yaw angle in every simulation.[331] Each simulation was run for 30,000 time steps, at 0.01 seconds per time step (i.e. 30 seconds of simulated behavior). For most collisions, the drones would complete their collision-avoidance maneuvers in less than 20 seconds.

To achieve this functionality, SwarmLab was extended in two ways. First, the class definition for the Swarm object was modified to permit multiple swarm instances with different parameters. For simplicity, the same control law, reference distance, and reference velocity was used for each swarm, but this need not have been the case. Second, SwarmLab's "spherical obstacle" code was extended so that each drone would perceive

[331] While analyzing the data for CHAPTER 6, it was observed that many of the model errors tended to cluster around the zero heading. This probably occurred because in all 5,000 simulations, the "red" flock was always initialized flying due north, while the blue flock was positioned all around the domain in order to collide with the red flock. This introduced a bias in the data, which translated into symbolic regressions that favored small final headings. That problem is eliminated in this chapter.

drones from the opposing swarm as moving spherical obstacles. The drones are only aware of the obstacles size and position. They have no sense of the obstacle's velocity. The radius of the obstacles was set as twice the collision radius of the drones, so that the drones would begin maneuvering earlier (the drone collision radius is 0.5m by default).

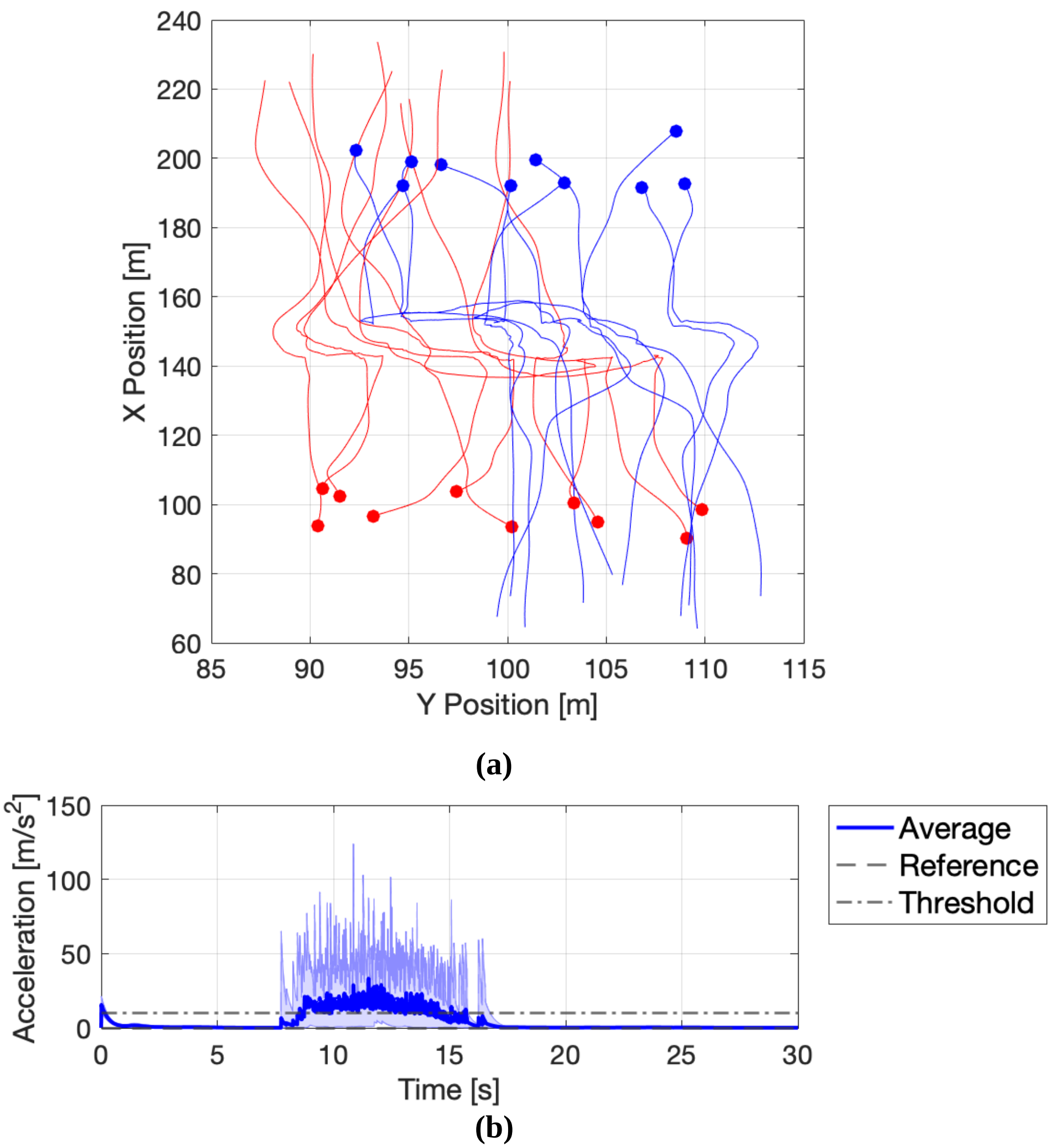


Figure 99 – (a) Sample drone trajectories for two-swarm glancing collision (b) drone acceleration time series indicating collision beginning ~7.5 seconds

Figure 99a shows the trajectories of the red and blue swarms that experienced a glancing collision. The dots indicate the starting positions of the drones. The circle one-another in a clockwise rotation before resuming their initial trajectories. Figure 99b shows the spike in drone acceleration that occurs at the onset of the collision. The data in Figure 99b is for the red swarm, but both swarms show the same characteristic behavior in every simulation. This information was used to detect collisions.

The output variables tracked in this behavior association test will be the changes in swarm speed (ΔS), pitch (ΔH_p), yaw (ΔH_y), shortest path length (ΔL_{sp}), principal axis length (ΔL_{pa}), and average drone separation (ΔD_{sep}). In order to avoid obtaining coupled functions, these outputs will be related to the initial values of the swarm and/or drone properties. For models where the only properties taken into consideration are swarm-level properties (referred to as swarm versus swarm, or SvS), the input properties are the initial values of the aforementioned outputs for both swarms. These properties are listed in Table 9. Note that the "1st" swarm is the swarm whose change is being measured, and the "2nd" swarm is the opposing swarm.

Table 9 – Swarm versus swarm input properties

S_1, S_2	$H_{p,1}$, $H_{p,2}$	$H_{y,1}$, $H_{y,2}$
$L_{sp,1}$, $L_{sp,2}$	$L_{pa,1}$, $L_{pa,2}$	$D_{sep,1}$, $D_{sep,2}$

Table 10 – Swarm versus drone input properties

S_1	$H_{p,1}$	$H_{y,1}$
$L_{sp,1}$	$L_{pa,1}$	$D_{sep,1}$
$S_{d,2}$	$H_{p,d,2}$	$H_{y,d,2}$

The initial time is the time step during which any drone in a given swarm first accelerates in response to the opposing swarm. The time interval ends when the swarm re-stabilizes. For models where individual drone properties (from the opposing swarm only) are used as inputs (swarm versus drone, or SvD) the properties are the initial Speed, Pitch, and Yaw of the drone (denoted with the subscript d, in the shaded row) as well as the initial properties of Swarm 1. Note that any model exclusively containing Swarm 1 initial properties as inputs must be a spurious regression.

8.2.2 Results

As shown in Table 11, nearly every property considered was indicted by the numerical criteria to be an emergent behavior. That is, the equations describing the effect that swarm level properties have on other swarm-level properties are less complex (CT) and have lower fitting error (RMSE) than equations describing the effect that properties from a randomly selected drone of the opposing swarm will have.

Table 11 – Summary of Pareto Front comparisons for swarm interactions

ΔS	ΔH_p	ΔH_y	ΔL_{sp}	ΔL_{pa}	ΔD_{sep}
+	-	+	+	+	+

Upon closer inspection, however, the sole Pareto optimal result for pitch that renders the decision a red "-" is an invalid interaction equation.

An interesting result (see Figure 108) is obtained for the change in average drone separation. No SvD models were obtained for this property because their extrapolation errors were massive. This means that SISSO could not find any drone properties that improved its regressions beyond a spurious regression. Only SvS models improved the prediction of ΔD_{sep} above the spurious regressions. While this does not undo the falsification result for Hypothesis 2, it does indicate that the underlying concepts and measurements have merit. The Pareto optimal equation is given by Eq. (68):

$$Dsep_{1,f} - Dsep_{1,0} = -0.204 - 2.816 \frac{S_{1,0}}{Dsep_{2,0}} \sin\left(Dsep_{1,0}\right) \quad \text{SvS:} \tag{68}$$

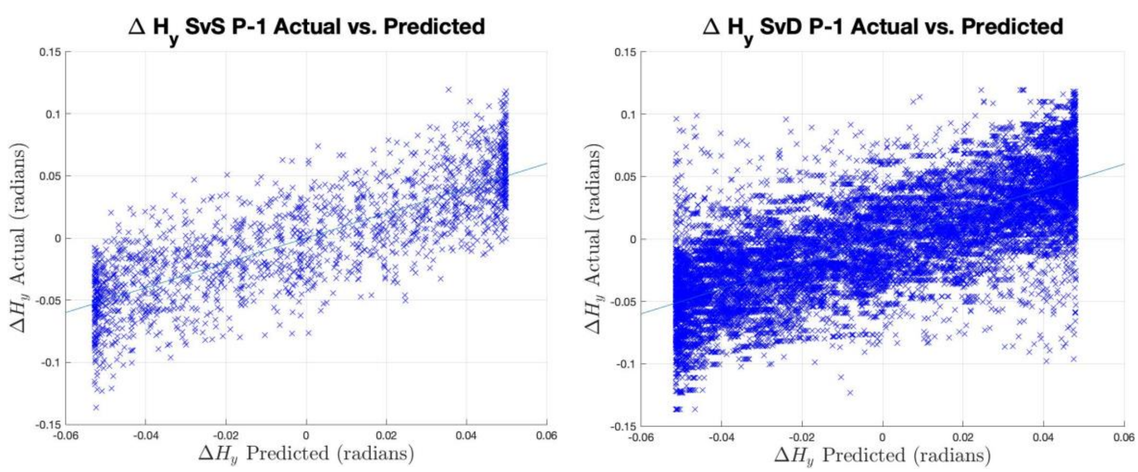
Recall that the subscripts 0 and f in Eq. (68) stand for initial and final, respectively.

The models for yaw show striking similarities (some being nearly identical),

$$\text{SvD P-1:} \quad Hy_{1,f} - Hy_{1,0} = -0.0017 + 0.050 \sin\left(Hy_{1,0} - Hy_{d2,0}\right) \tag{69}$$

$$\text{SvS P-1:} \quad Hy_{1,f} - Hy_{1,0} = -0.0015 + 0.051 \sin\left(Hy_{1,0} - Hy_{2,0}\right) \tag{70}$$

As shown by the Actual versus Predicted plots (Figure 100), the errors of these models are much better behaved than the errors obtained for the Flocking Vee model.



(a) **(b)**

Figure 100 – Actual versus Predicted plots for (a) Eq. (69), (b) Eq. (70)

In fact, many Pareto optimal models appear to be excellent fits. Figure 101 shows examples of speed, and pitch. This dramatic improvement is undoubtedly due to the stability of the swarms (as enabled by their control laws), and the swarm's ability to return to its mission (maintain speed and heading) despite encountering a moving obstacle. Comparing pitch to yaw, it is clear that the level collisions destabilized drone yaw much more than pitch.

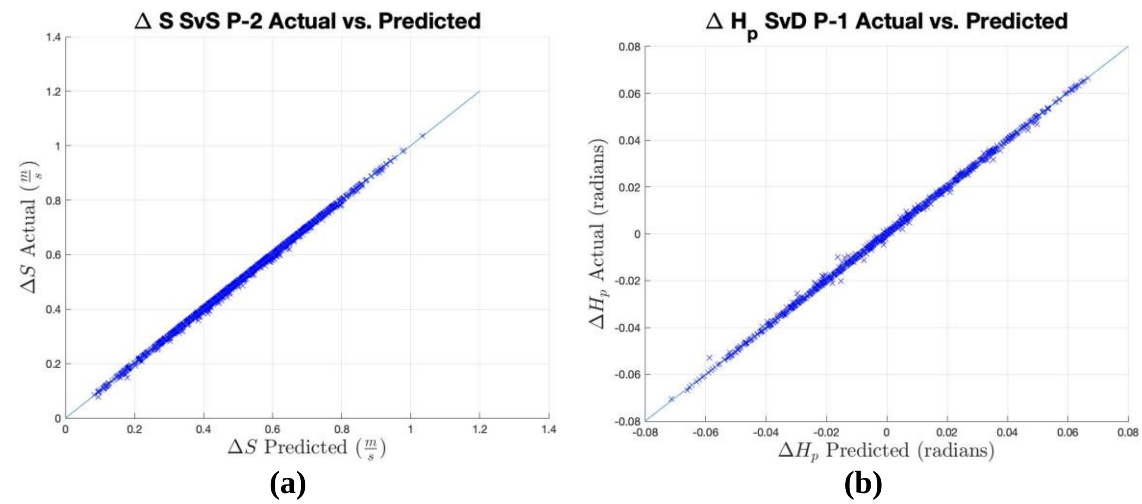


Figure 101 – Sample Pareto optimal regressions for (a) swarm change in speed, (b) swarm change in pitch

The distance metrics regressions fared worse (see Figure 102). The primary cause is probably that these metrics experienced much larger relative changes than the swarm's speed and heading since the control law does not strictly enforce values for these properties.

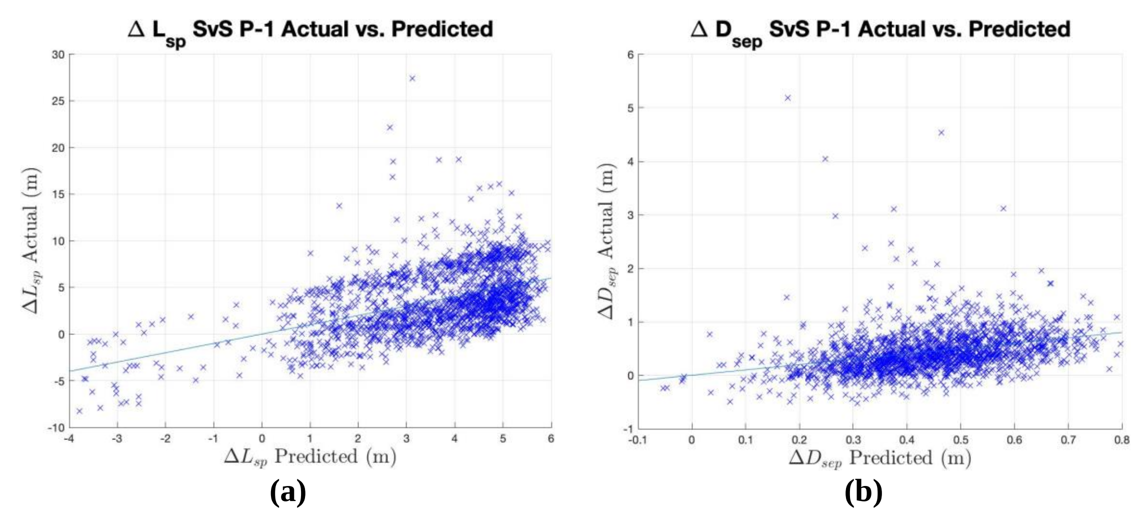
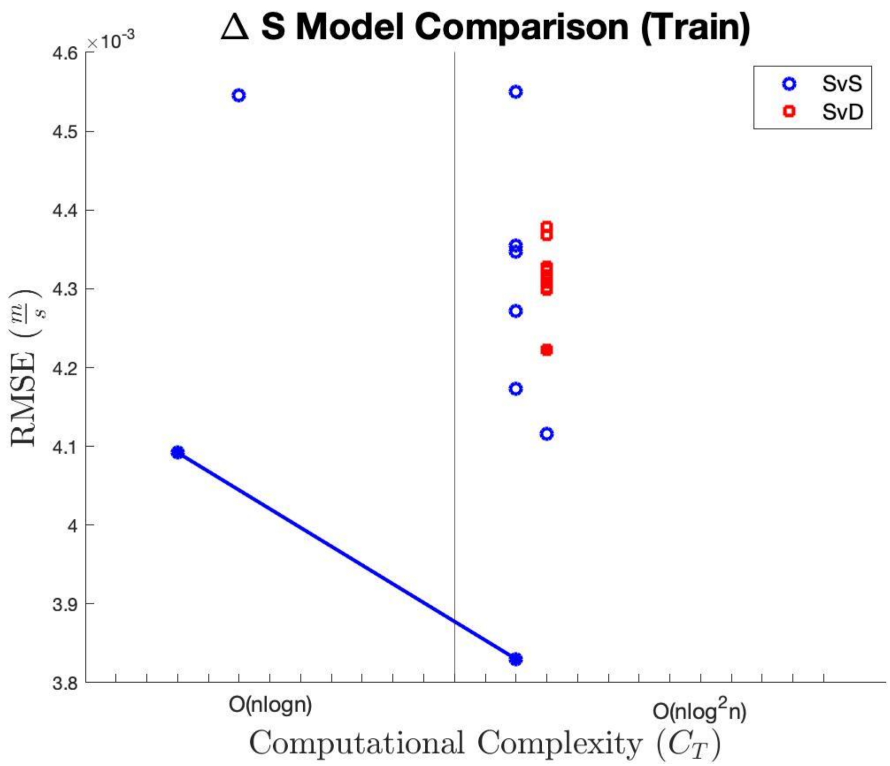
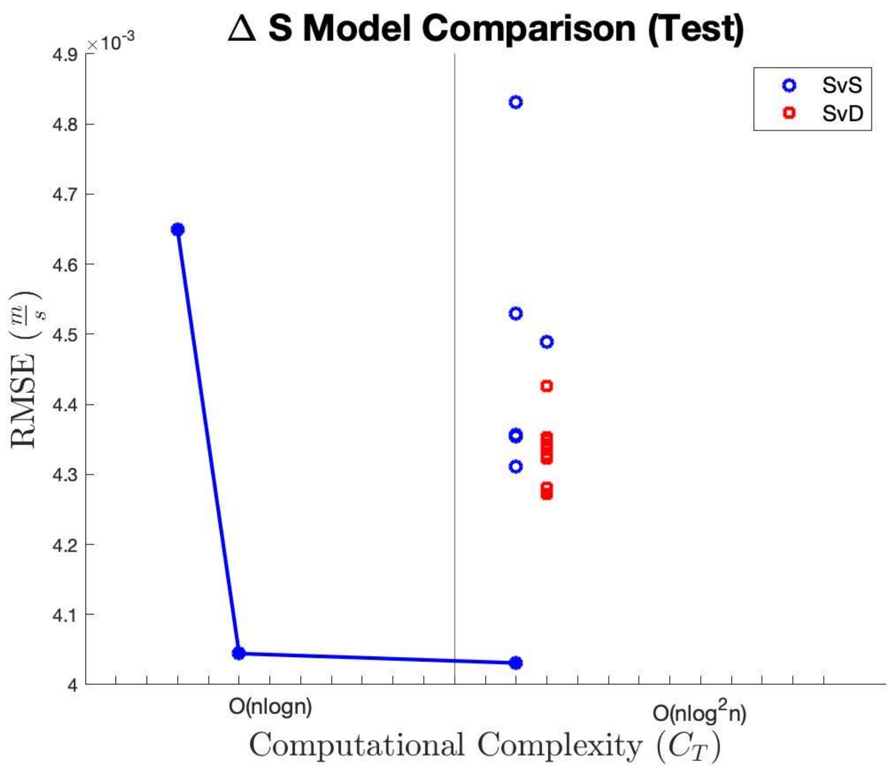


Figure 102 – Actual versus Predicted plots for a Pareto optimal model of (a) change in shortest path length, (b) change in average drone separation

Considering that the errors in Figure 101b are for a model built from the data of a randomly selected opposing drone, it is clear that the control laws had a significant effect on the variability of some properties and, by extension, the goodness of fit of the regressions.



(a)



(b)

Figure 103 – Pareto optimal swarm Δ speed models due to interaction and re-stabilization (a) training data (b) test data

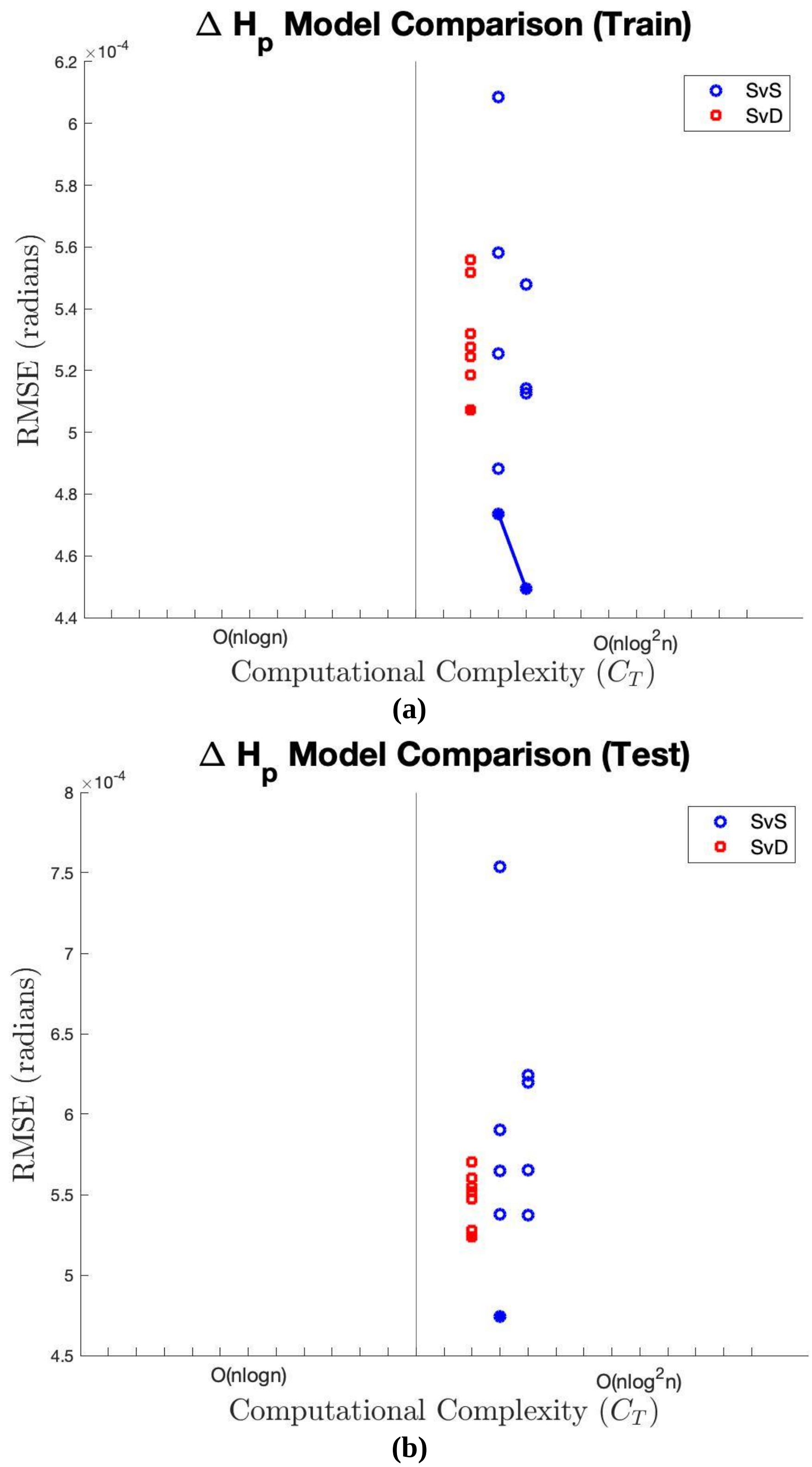
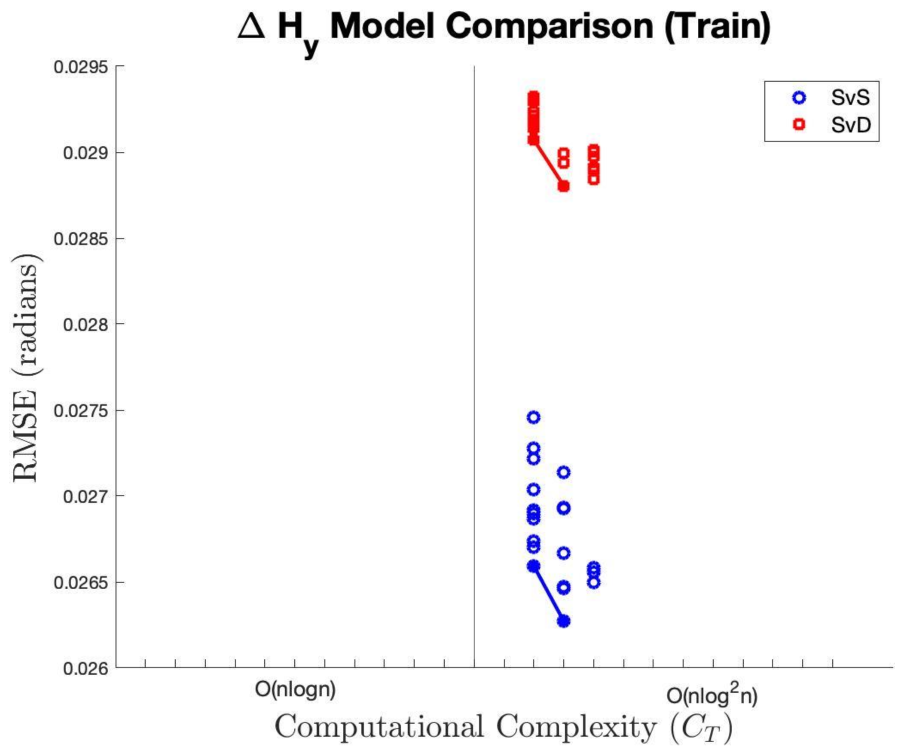
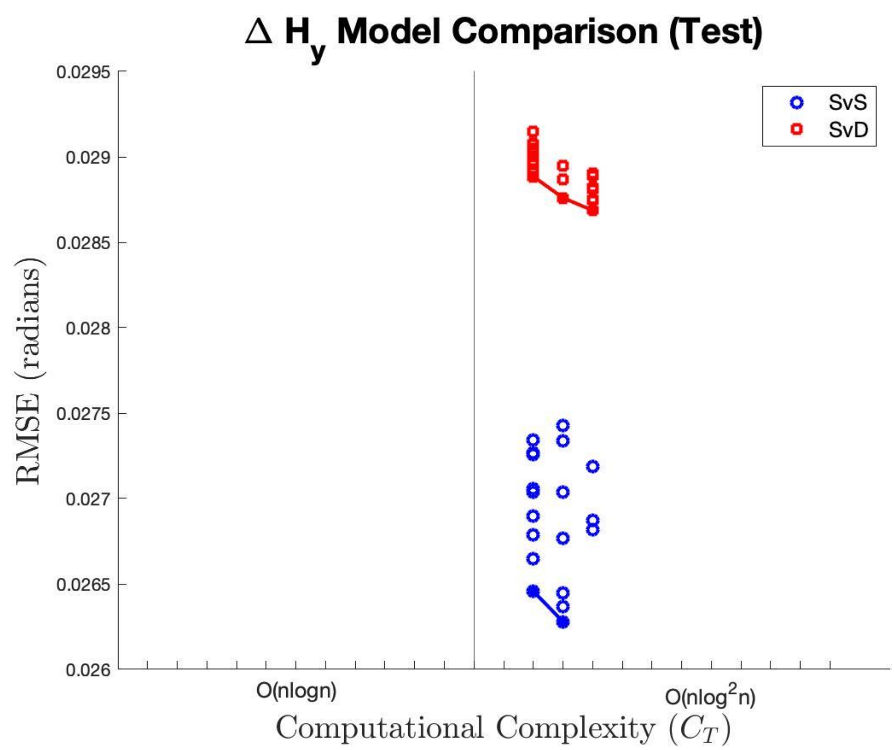


Figure 104 – Pareto optimal swarm Δ pitch models due to interaction and re-stabilization (a) training data (b) test data

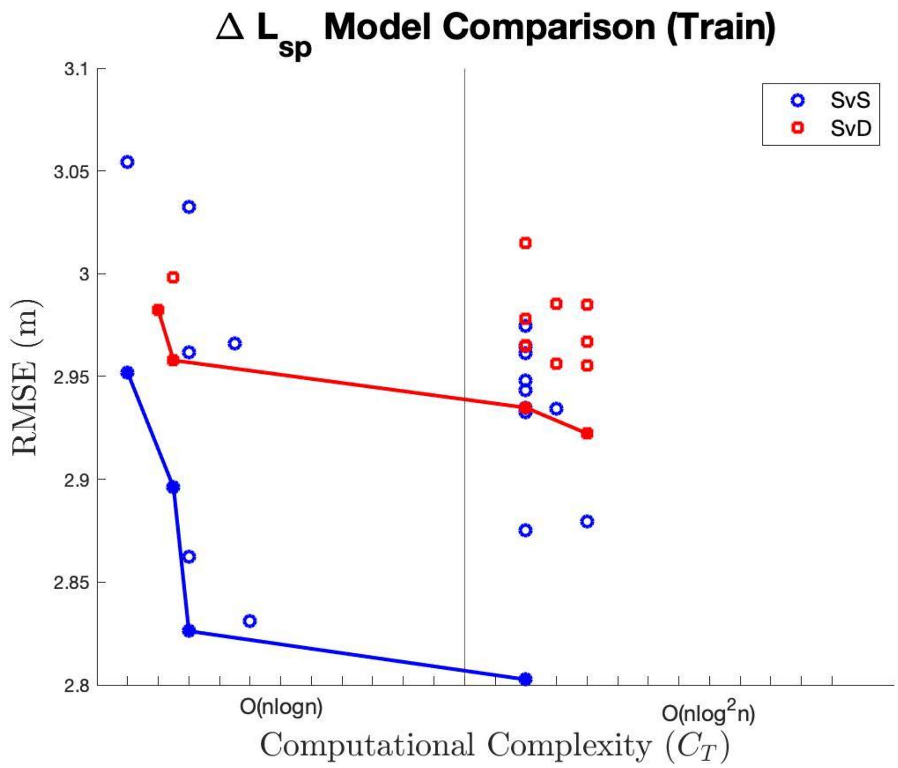


(a)

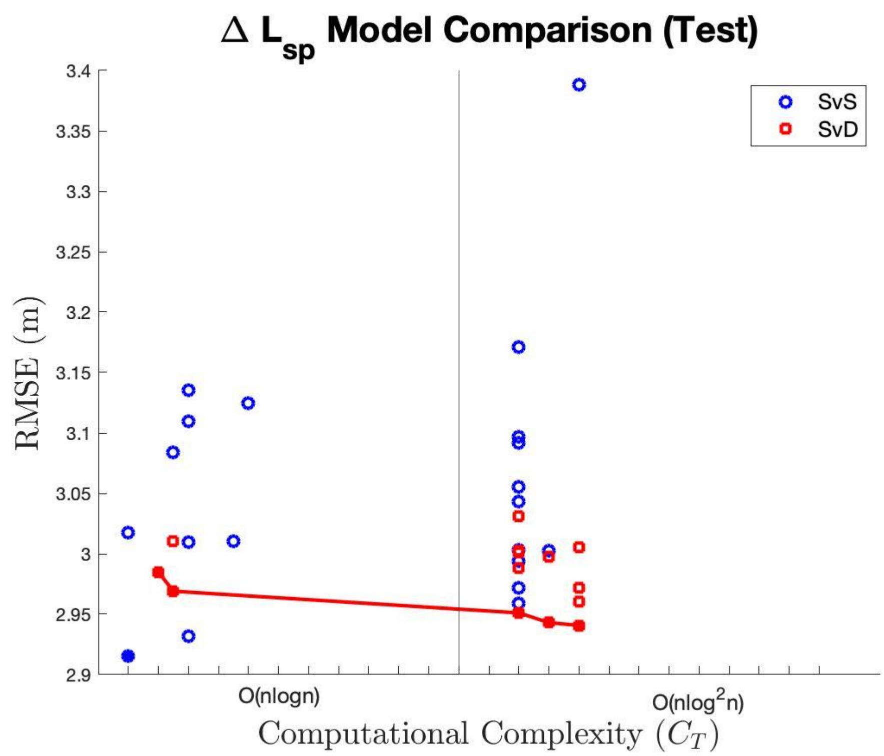


(b)

Figure 105 – Pareto optimal Δ yaw models due to interaction and re-stabilization (a) training data (b) test data

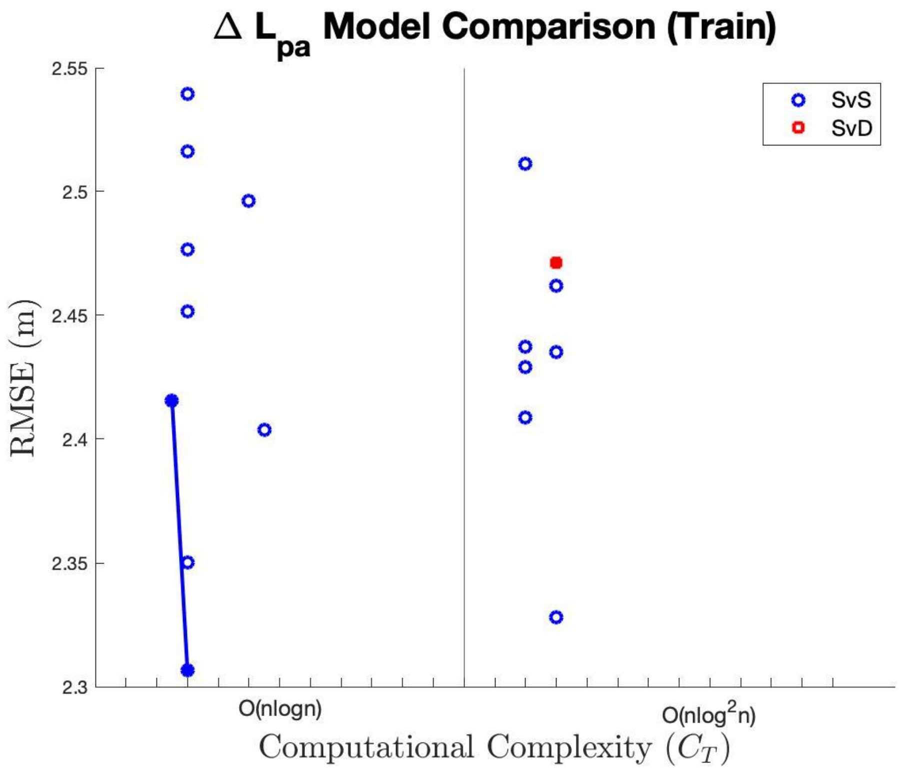


(a)

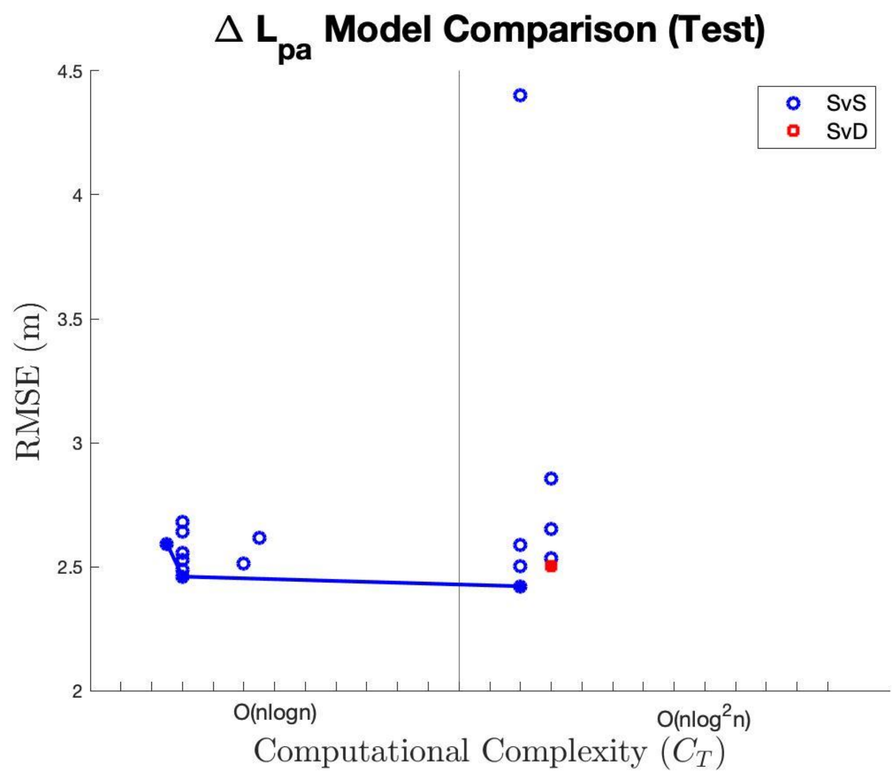


(b)

Figure 106 – Pareto optimal swarm Δ shortest-path models due to interaction and re-stabilization (a) training data (b) test data

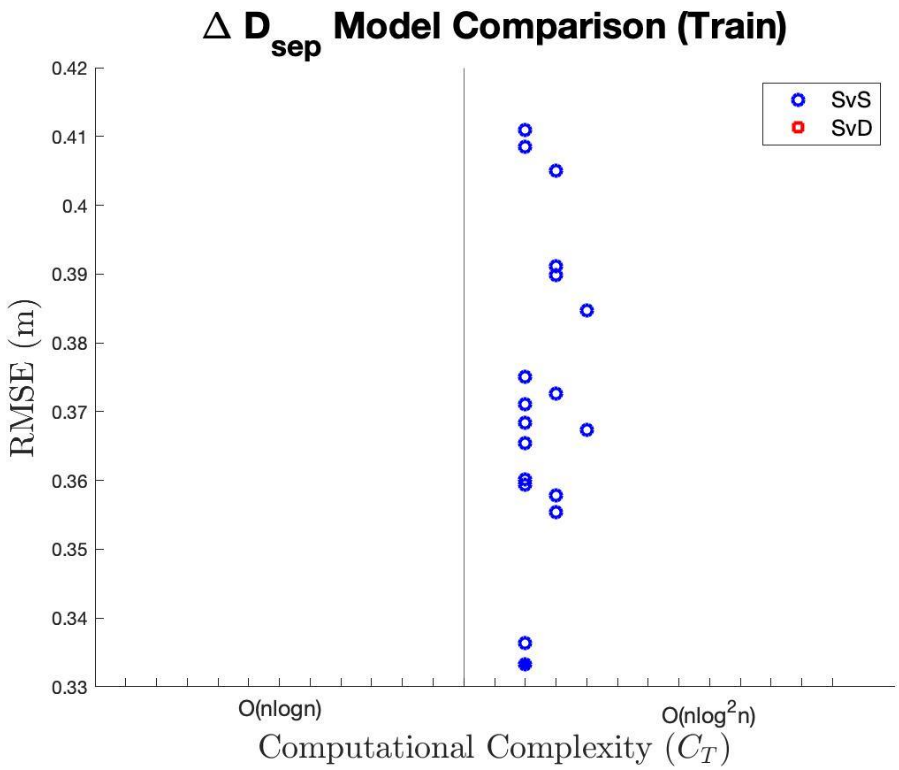


(a)

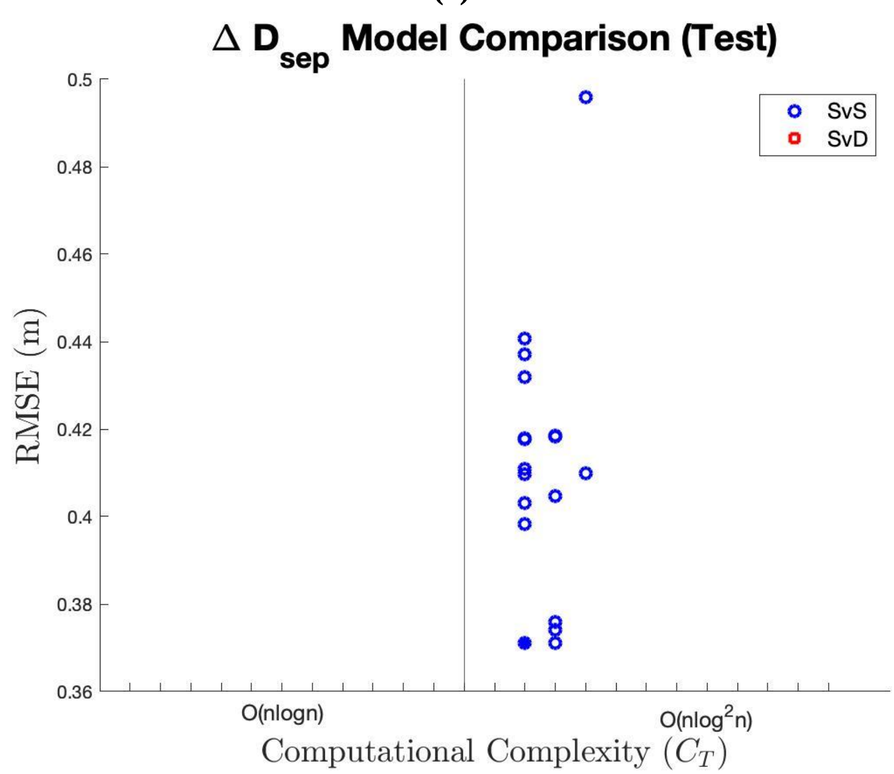


(b)

Figure 107 – Pareto optimal swarm Δ length of principal axis models due to interaction and re-stabilization (a) training data (b) test data



(a)



(b)

Figure 108 – Pareto optimal swarm Δ average drone separation models due to interaction and re-stabilization (a) training data (b) test data

8.3 Exploitation Analysis

For this section, SwarmLab is extended once again so that one swarm's mission will be geared toward the manipulation of the opposing swarm's properties. Rather than simply being instructed to maneuver from one fixed location to the next, the swarm's direction and speed will be determined by some mission objective. For the first maneuver, the adversarial swarm will remain intact, while in the second maneuver, it will split into two sub-swarms which will then act in concert to affect the opposing swarm.

8.3.1 Measures of Merit

SwarmLab measures five different swarming-specific performance metrics: (1) order, (2) safety among drones, (3) safety with respect to obstacles, (4) union, and (5) connectivity. Order is simply the average of the normalized velocity dot products for all pairs of drones in the swarm (an aggregate of the same metric used thus far in self-organization detection). Safety is measured by counting the number of times drones risk actual collision (by flying too close to another drone/obstacle) and dividing that by the total number of pairs of drones, or the total number of obstacles, respectively (i.e. number of all possible collisions). The union metric tracks the number of independent subgroups that form during a simulation [254]. The connectivity metric is the algebraic connectivity of the graph corresponding to the swarm and is calculated by dividing the second smallest eigenvalue of the Laplacian matrix by the number of drones [254]. An example of the performance analysis time series generated by SwarmLab for a single simulation is given in Figure 109. In this example, most performance metrics are equal to one.

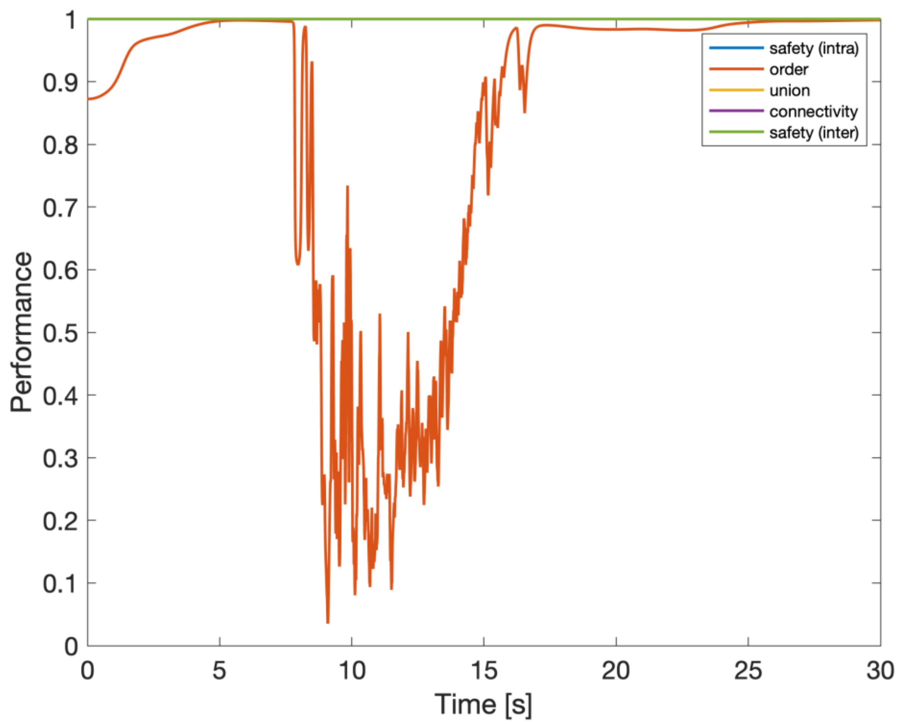


Figure 109 – Sample performance analysis plot generated using SwarmLab

The order of the flock changes dramatically during a collision, as each drone maneuvers to avoid its nearest obstacle (typical of interaction and re-stabilization time intervals).

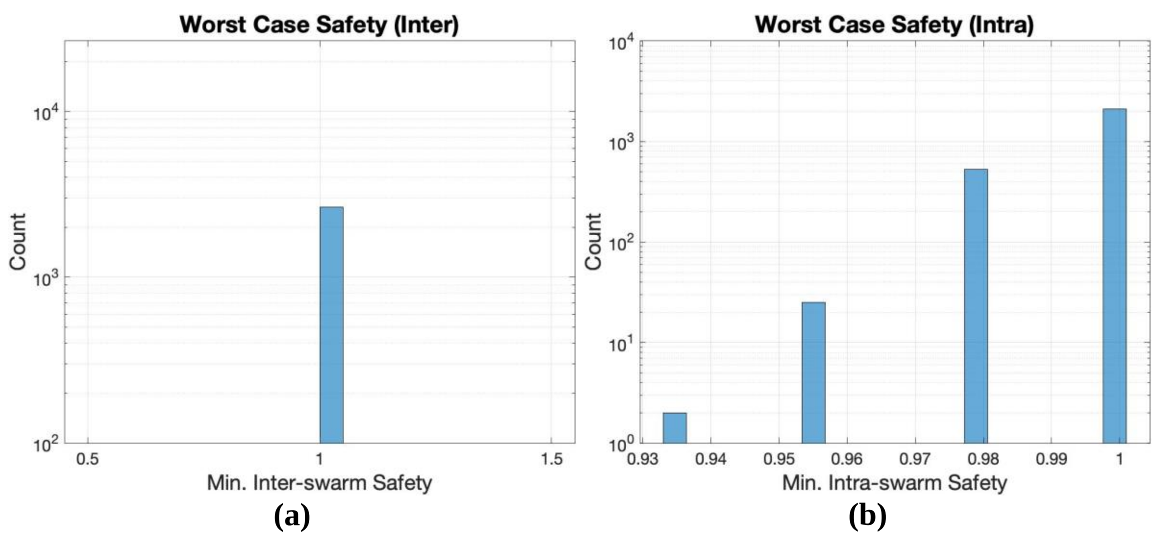


Figure 110 – Minimum safety values for swarm collisions in Section 8.2 (a) inter-swarm collisions, (b) intra-swarm collisions

Histograms summarizing the lowest safety value obtained at any time during simulation for all simulations executed in Section 8.2 are shown in Figure 110. Figure 110a shows that, for all 1,326 simulations, there were no collisions among drones of opposing swarms. Within each swarm, however, Figure 110b shows that in 20% of cases, the safety metric equaled 0.9778 (1 collision[332]), while in 0.94% cases the safety metric equaled 0.9556 (2 collisions), and in very rare cases the safety metric equaled 0.9333 (3 collisions).

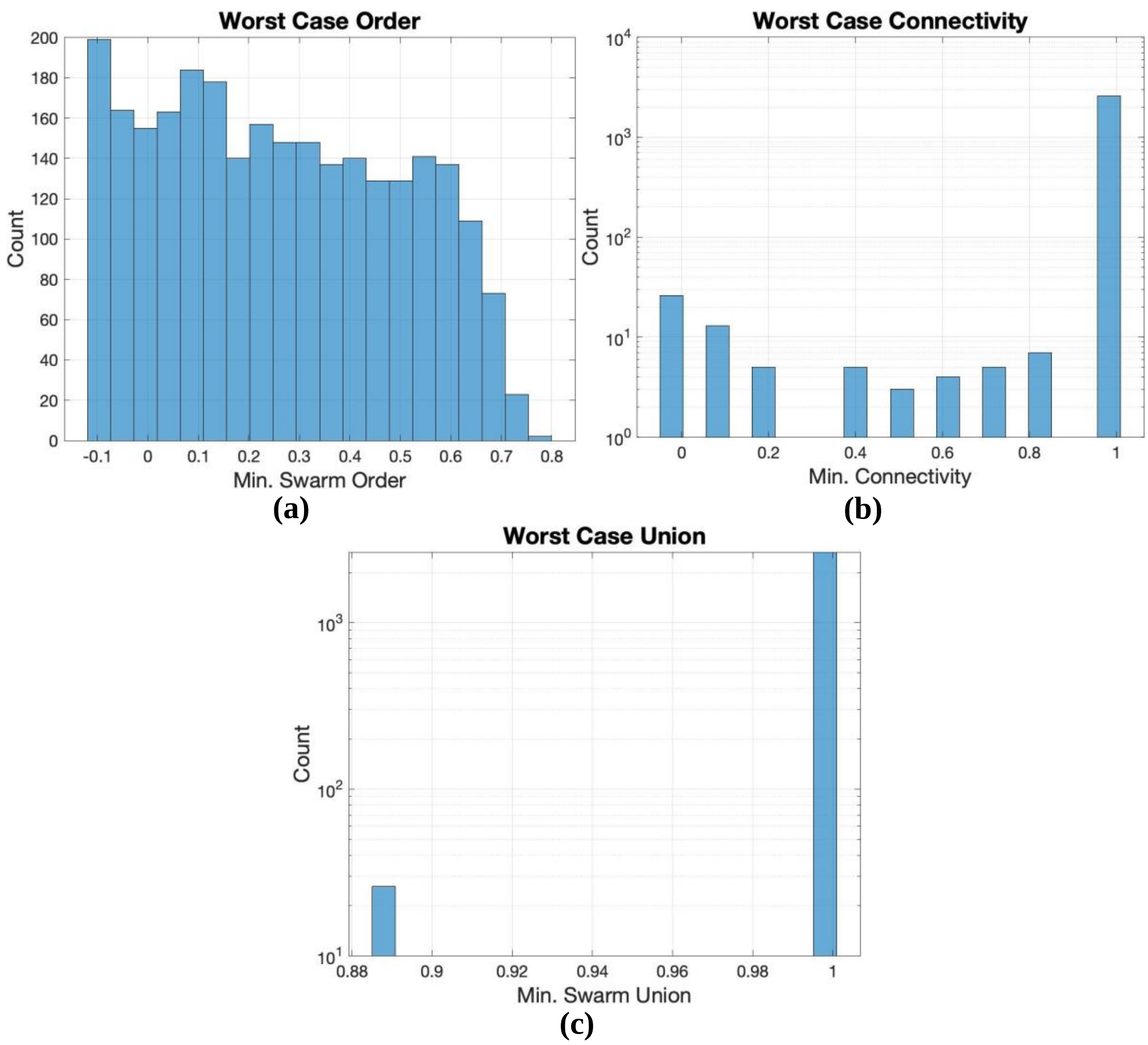


Figure 111 – Histograms of performance metrics (worst-case value per simulation) for swarm collision simulations in Section 8.2 (a) order, (b) connectivity, (c) union

[332] Collision values are based on a 10-drone swarm.

Figure 111a clearly shows that the swarm's order is affected by collisions in every simulation (as expected). Some very severe head-on collisions even cause the majority of the drones in the swarm to briefly reverse course (negative order values). The union is rarely affected by collisions, and less than 5% of all collisions affect the swarm's algebraic connectivity. These histograms will be used as reference values for the results in Sections 8.3.4-8.3.5.

8.3.2 Sensitivity Analysis

There are a number of goals one could consider for a swarm. One might choose to perform a search mission, in which case the swarm in question is under the control of the user, and that user might consider how changing the drone's actuators or the swarm's control law would affect its search performance. However, the drones simulated here are point-masses (no actuators). Furthermore, arguing that a modification to a control law to induce a change in an emergent behavior would risk devolving into a circular argument under the definitions and methods of this thesis. Therefore, this thesis will instead consider an adversarial swarm mission, where one swarm must be directed to affect the swarm-level properties of the opposing swarm. This is compatible with the definition of weak emergence. Furthermore, if the resulting behaviors result in a significant change to a swarm's measures of merit, that information can be used to infer a purpose (functional emergence). For example, if it can be shown that a particular behavior decreases a swarm's safety, that behavior could have combat applications. In the case of an adversarial mission, every sensitivity analysis suggests that a "ways" solution is needed. However, after deriving some new behavior to serve that purpose, it may become obvious, for example, that the drones cannot move fast enough to achieve the desired maneuver. This would then imply that a "means" solution is also needed (some technology improvement, etc.).

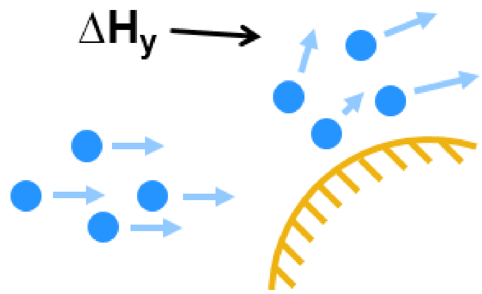


Figure 112 – Example of change in swarm yaw (solid blue circles represent drones) due to obstacle (yellow curve)

The first property to consider is the yaw of the swarm (H_y), given by the equations:

$$\vec{V}_{swarm} = \frac{1}{n}\sum_{i=1}^{n}\vec{V}_{drone,i} \tag{71}$$

$$H_y = \tan^{-1}\left({V_{swarm,y}}\big/{V_{swarm,x}}\right) \tag{72}$$

Where $V_{swarm,y}$ is the y component of the velocity. The time-derivative of H_y is given by,

$$\frac{dH_y}{dt} = \frac{1}{\left\|\vec{V}_{swarm}\right\|^2}\left(V_{swarm,x}\frac{dV_{swarm,y}}{dt} - V_{swarm,y}\frac{dV_{swarm,x}}{dt}\right) \tag{73}$$

As Eq. (73) indicates (as does common sense), the only way to affect a swarm's yaw is by compelling it to turn. In the case of this simulation, it has already been observed that swarms turn when confronted with an obstacle, therefore, the adversary swarm must behave like an obstacle. However, it has also been observed that if a swarm is given the mission of flying along a particular route, it will resume its original heading after clearing the obstacle. Therefore, the adversary swarm must become a persistent obstacle, which indicates that the adversary swarm must also move along with the opposing swarm in order to guide it along a heading that it did not intend to go. This immediately suggests a function, since changing a swarm's course can cause it to fail its mission, or benefit its adversary.

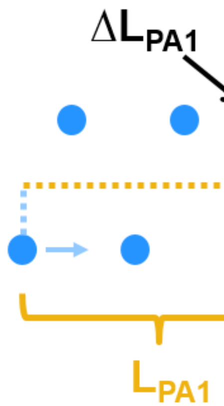


Figure 113 - Example of change in L_{PA1} by displacing drones (solid blue circles) at either extreme of a six-drone swarm

The second property to consider is the swarm's largest principal axis length (L_{PA1}). Obtaining this value, in general, requires performing a principal component analysis to obtain a matrix with which to rotate the point cloud of swarm positions to match the reference coordinate system. Afterwards, the following equation can be applied:

$$L_{PA1} = \max(P_{drone,i,PA1}) - \min(P_{drone,j,PA1}) \ \forall \ i,j \in [1,n] \tag{74}$$

Where $P_{drone,i,PA1}$ is the component of the position of the ith drone on the principal axis (pa), and n is the number of drones in the swarm. For a Cartesian coordinate system, the largest principal axis is typically set to be the x-axis. To change L_{PA1} one must move either drone on either extreme either away from the swarm, or back towards its center.

$$\frac{dL_{PA1}}{dt} = \frac{d}{dt}\left(\max\left(P_{drone,i,PA1}\right)\right) - \frac{d}{dt}\left(\min\left(P_{drone,j,PA1}\right)\right) \forall\, i,j \in [1,n] \qquad (75)$$

Eq. (75) is provided for completeness. Since the swarm control law discourages individual drones to rush ahead of the swarm, it seems the most effective course of action for an adversary swarm would be to compel opposing drones to fly back towards the center of the swarm (as in Figure 113). This process can be repeated one drone at a time, but that raises a variety of issues. Firstly, said approach reduces to an inefficient and potentially useless game of whack-a-mole. Secondly, and related to the first reason, depending on the distribution of drones, the principal axis may rotate[333] faster than it shrinks, which complicates the maneuver required to shrink it. Thirdly, if the maneuver is set up to affect one drone at a time, then the system-level interaction has become an indirect effect, rather than the direct cause (in engineering terms). Philosophically, of course, the drone-level interactions must and will always occur in order to achieve swarm-level changes.[334] However, the overarching goal of this thesis is to develop a method for exploiting system-level properties, which includes controlling system-level behaviors. Therefore, consider the consequences of an entire swarm affecting the principal axis of an opposing swarm: the opposing swarm will change shape. Then consider what would happen if the opposing swarm's principal axis were affected from both directions: the swarm would be pinched. Since the swarm doing the pinching is an amorphous cloud, it stands to reason that said

[333] This would be more obvious had the time derivative been taken on the full equation in the original coordinate system, since it would include a rotation matrix from the coordinate transformation.

[334] The extent to which altering, removing, or otherwise affecting a single component impacts the whole system depends on how tightly coupled the whole system it, how de-stabilizing the effects are, and how well the remainder of the system re-stabilizes after being perturbed. This thesis has not explored the relationship between "how coupled" a system's components are, and "how complex" the system is. This is left for future work. It is worth noting that the flock systems of CHAPTER 6 are far less coupled than the pursuit systems of CHAPTER 7 and the swarms in this chapter.

swarm can pinch the opposing swarm such that it becomes roughly planar ("sandwiched").[335] Such a change in structure would require significant maneuvering on the part of the opposing drones, and would test the limits of the control law governing them. In this case, the focus is less on shrinking the principal axis for its own sake, and more on deforming a swarm. The maneuver required to do so is simply a more forceful and persistent version of the maneuver required to shrink the principal axis.

8.3.3 Simulation Settings and Modifications

As in Section 8.2, two swarms of UAVs will operate in an obstacle-free environment. A 10-drone swarm (blue) will be instructed to fly at a constant heading (yaw) across the domain, while a 10-drone swarm (red) will be instructed to intercept the blue swarm with the goal of affecting either its H_{yaw} or L_{PA1}. Both swarms have an upper velocity threshold of 7m/s, but the blue swarm will cruise[336] at 5m/s while the red swarm will cruise at 6m/s. Both swarms will have an acceleration threshold of 10m/s^2, but since the drones are modeled as point-masses, the drones will routinely violate this threshold (this can be safely neglected). Both swarms will follow the OSM control law, and have a reference distance of 10m. All drones will have a vision radius of 100m, and a collision radius of 0.5m. For maneuvering purposes, drones of the opposing swarm will be treated as though they have a radius of 1m (i.e. the spherical obstacle radius is 1m).

Red swarm's mission will be specified entirely by changing its "migration direction" and cruise speed. In other words, the control law is completely preserved. The

[335] That is, shrinking the original largest principal axes until the variability along that axis is very small compared to the variability along the other two axes. The resulting swarm is effectively two-dimensional.
[336] Its reference velocity (p_swarm.u_ref).

only change to swarm behavior is provided through a single velocity term, which is the term in the control law that informs the drones of the direction in which they "should" go and how fast (provided there is no risk of collision). A total of 1,200 25-second simulations will be run for H_{yaw} and 1,200 34-second simulations will be run for L_{PA1}.[337] The time step sizes are 0.01 seconds for all simulations. The changes in H_{yaw} and L_{PA1} reported below are based on time-averaged initial and final values. The initial values is the average of the property over a one-second time interval ending just before the swarms interact. The final value of H_{yaw} is the average angle of the flock trajectory over the last second of the simulation. Note that this is not the same as the average heading of the drones, computed from their velocities. That is because the control law continues to attempt corrections of drone headings. Therefore, although the drone headings may average out to match their mission heading at a given time step, the net movement of the swarm over time will not reflect that heading. The final value of L_{PA1} is the average property value over a one second interval that begins/ends ±0.25 seconds from the instant at which the blue flock velocity along the horizontal plane reaches its minimum value (i.e. the blue flock cannot advance because it is being pinched). The time interval is shortened because the blue swarm drones move rapidly to avoid the incoming obstacles. This time interval is a slight deviation from the interaction + re-stabilization time interval previously discussed in this thesis. There is a brief moment of meta-stability when the swarm is nearly arrested, and just before it attempts to avoid the red swarm, where the L_{PA1} can be measured to determine if the red swarm has achieved its mission. Technically, however, since the blue swarm is commanded to continue advancing, that brief stability gives way to another interaction phase, meaning

[337] There were 84 L_{PA1} cases where the swarms did not complete the maneuver in time. These cases were dropped from the results.

that the blue swarm has not had the opportunity to fully re-stabilize before responding to the next interaction phase. Another deviation from the previous examples in this thesis is that the swarm is divided into two halves and manipulated separately. This should be thought of as a first step towards generalization of the previous cases to SoS and meta-stable systems.

8.3.4 Results: Swarm Yaw Case Study

In order affect the blue swarm's yaw, the red swarm is directed to fly as follows: (1) if either shortest path to intercept the opposing swarm can be reached at a time greater than zero and the separation between the two swarms is greater than a minimum threshold, then move along the positive minimum-time-to-intercept path, (2) otherwise, fly to a point 11.5m ahead of the blue swarm along the trajectory of the blue swarm, match its heading, and fly at a speed 5% slower than the blue swarm speed. The second rule causes the swarm to behave like a persistent obstruction. This maneuver is depicted in Figure 114.

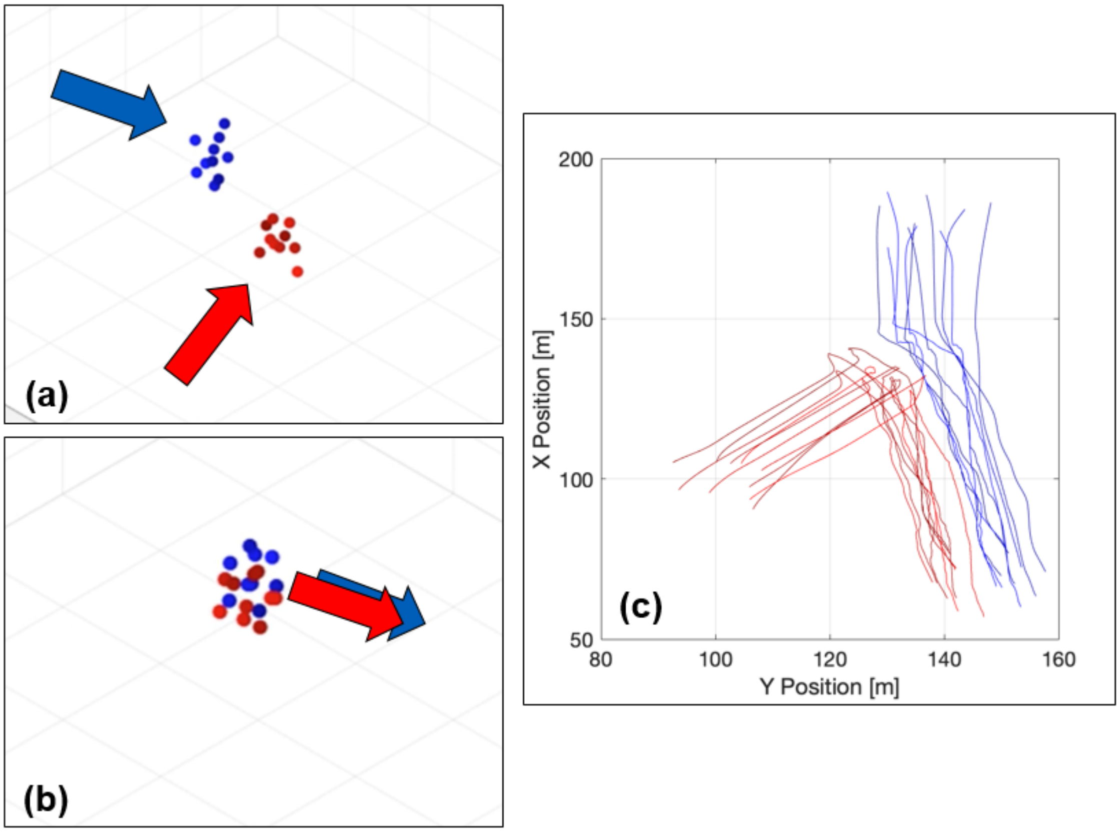


Figure 114 – Screenshots of adversarial swarm mission: change blue H_y. (a) Red en route to intercept blue swarm. (b) Red persistently pushing blue swarm. (c) Swarm trajectories indicating red nudging blue away from its vertical path.

As shown in Figure Figure 114a, the red swarm flies to intercept the blue swarm. Figure Figure 114b shows how the red swarm flies very closely to the blue swarm, gently nudging it off of its trajectory (the full trajectories are shown in Figure Figure 114c).

As indicated in Figure 118, the red swarm's maneuvering slightly increases the collision risk of its drones relative to the blue swarm, but there are no collisions between drones of opposing swarms. Figure 118c-d indicate that the red swarm can execute this maneuver without splitting into smaller sections, and the blue swarm retains its organized form.

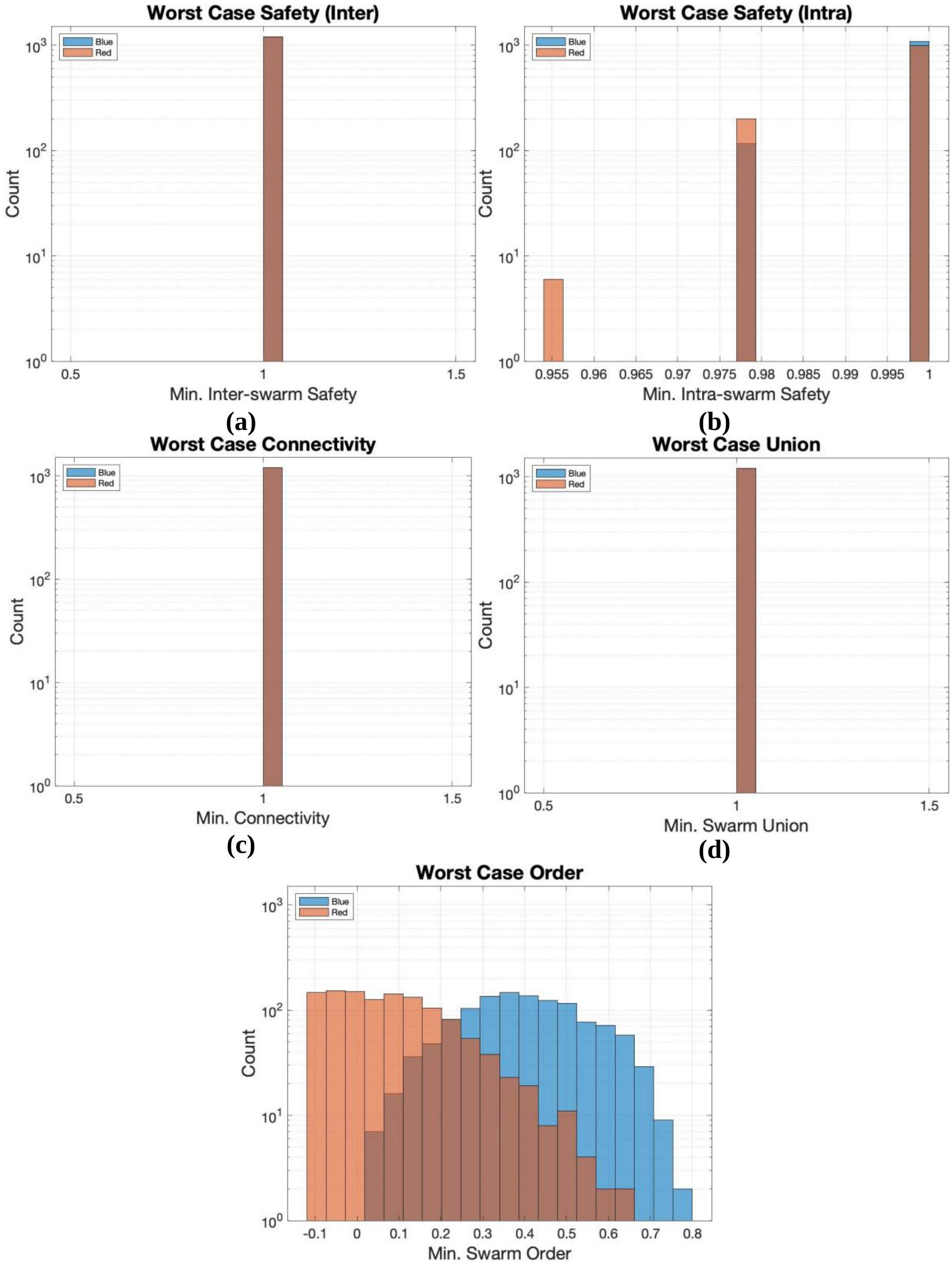


Figure 115 – Performance statistics (change H_{yaw})for red and blue swarms (a) inter-swarm collisions, (b) intra-swarm collisions (c) connectivity, (d) union, (e) order

Figure 118e indicates that the drones in red swarm do not fly as cohesively as the blue swarm's drones do, having to change directions more frequently and, in some cases, fly orthogonal to one-another or even in slightly anti-parallel directions. Overall, these results indicate that the risks of the maneuver are about as great for the blue swarm as they are for the red swarm, but that the red swarm would benefit from increasing the maneuverability of its drones (either by changing its control law, or improving drone performance).

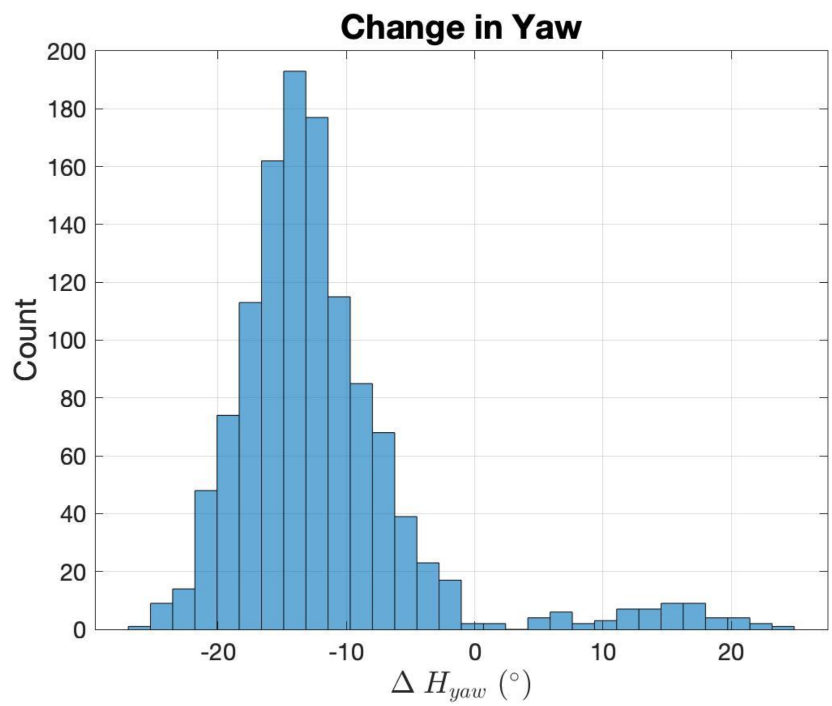


Figure 116 – Histogram indicating effectiveness at changing H_{yaw}

Figure 119 clearly indicates that the red swarm was largely successful at redirecting the blue swarm. In some cases, the very basic maneuver coded for this thesis resulted in deflections approaching 25°. Clearly, a more sophisticated maneuvering procedure can be implemented to cause the blue swarm to veer off course even more. This example shows that a simple sensitivity analysis can produce useful diagnostic information for means versus ways decisions. However, more prescriptive methods are needed.

8.3.5 Results: Swarm Principal Axis Length Case Study

The maneuver to affect the blue swarm's principal axis length is more involved. The red swarm is given instructions to coordinate their behavior as though they were two separate swarms. One sub-swarm follows the aforementioned minimum-time-to-intercept path to a point 25m ahead of the blue swarm and then matches the speed and heading of the blue swarm (see Figure 117a). The second sub-swarm maneuvers behind the blue swarm (giving it a wide berth, as shown in Figure 117a) until it reaches a location 15m directly aft of the blue swarm, and then matches its speed and heading (see Figure 117b). These lengths have been computed so that then the sub-swarms later pinch the blue swarm, they arrive at the same location at roughly the same time.

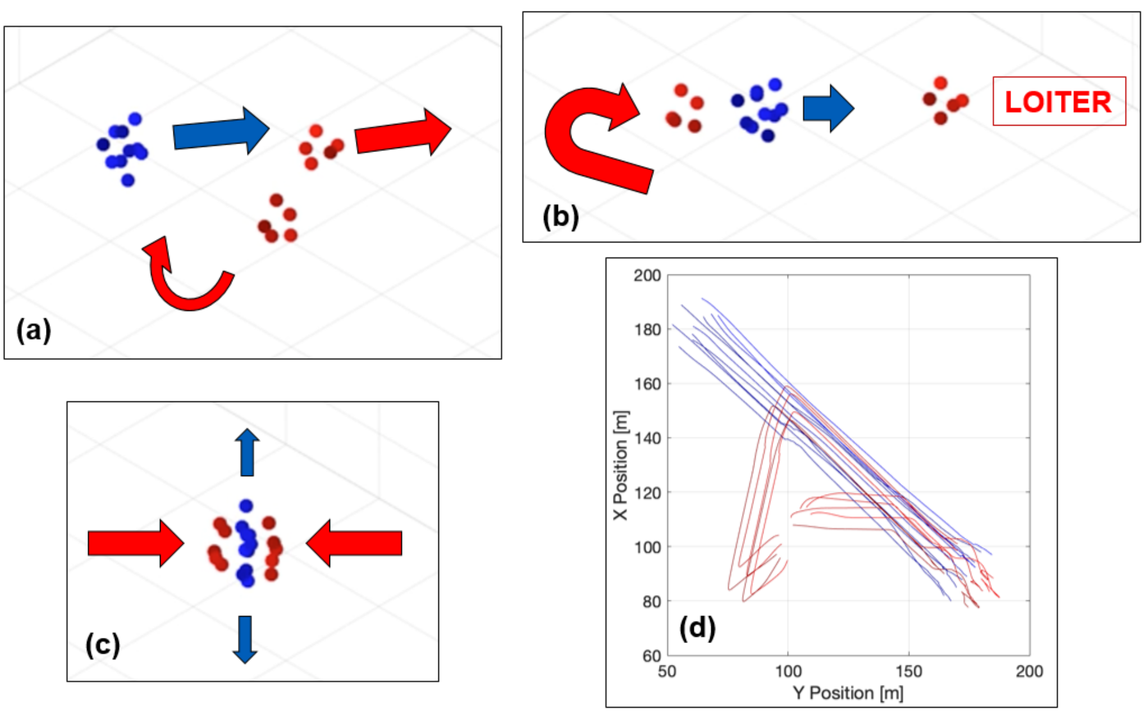


Figure 117 – Screenshots of adversarial swarm mission: change blue L_{PA1}. (a) Red swarm splits: half leads blue, half circles behind. (b) Aft red swarm catches up to blue while other half loiters. (c) Both red halves pinch blue. (d) Swarm trajectories.

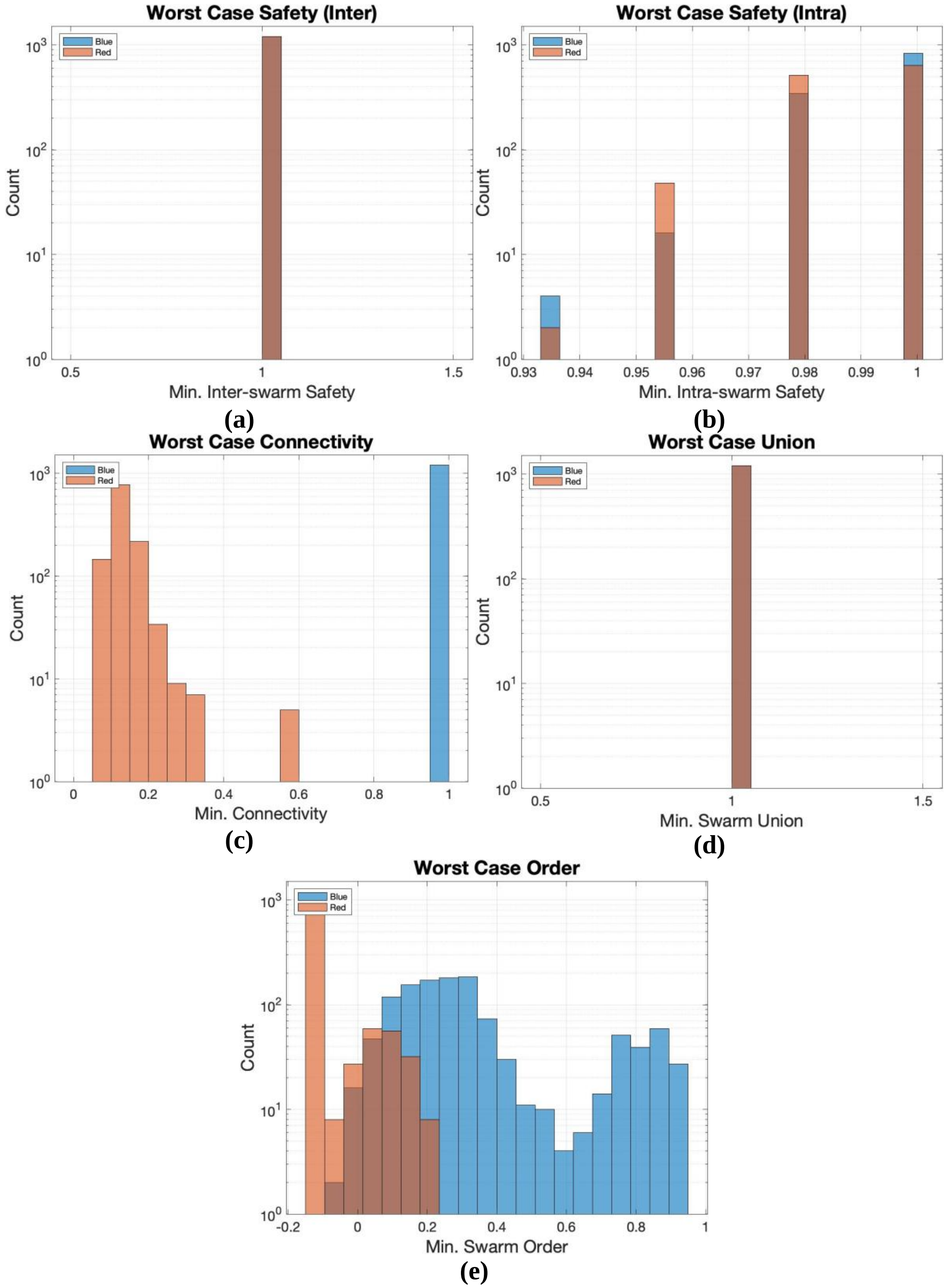


Figure 118 – Performance statistics (change L_{PA1}) for red and blue swarms (a) inter-swarm collisions, (b) intra-swarm collisions (c) connectivity, (d) union, (e) order

Once both sub-swarms report that they are matching the blue swarm speed and heading, the forward sub-swarm comes to a halt (see "loiter" in Figure 117b), while the second swarm accelerates into the blue swarm. Once the blue swarm is close to the loitering sub-swarm, that sub-swarm will travel full speed towards the blue sub-swarm, and vice-versa (thereby pinching the blue swarm between them as in Figure 117c). The trajectories corresponding to these maneuvers are indicated in Figure 117d.

As in Section 8.3.4, Figure 118a shows that there are no collisions between drones of opposing swarms, but within each swarms the drones do occasionally collide. While 13.4% of cases in Section 8.3.4 resulted in collisions (total for both swarms), here 38.8% of cases resulted in collisions. Once again, the increased risk is fairly evenly distributed. The connectivity shown in Figure 118c simply reflects that the two sub-swarms operate as separate units far apart from one-another, and the large spike in Figure 118e indicates that both sub-swarms fly away from one-another in several missions.

Figure 119 shows that in most cases, the L_{PA1} actually grew rather than shrank. This is likely caused by the axis rotating and stretching as the blue swarm is flattened into a horizontal configuration. While the sensitivity analysis did not prohibit this from happening (in fact, it was somewhat expected), the results of that analysis did not indicate how to avoid such a behavior. Thus, again, a more prescriptive method is needed. If one considers the shortest principal axis, L_{PA3}, rather than the longest it is clear that the red swarms succeeded in compressing the blue swarm into a planar configuration in the majority of cases, but in 16.4% of cases, the maneuver had no effect, or the blue swarm broadened slightly. With respect to L_{PA3}, the swarm performed very well (see Figure 120).

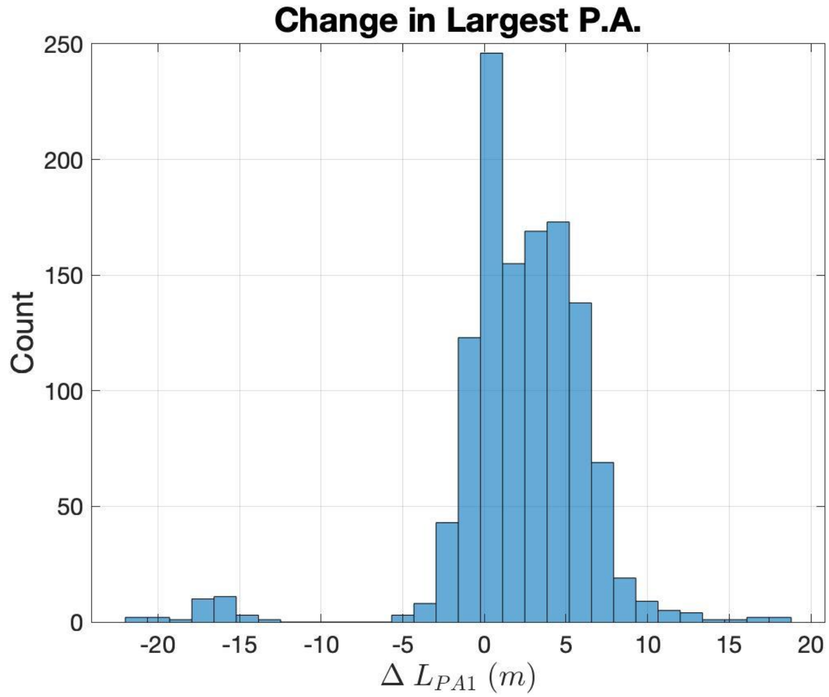


Figure 119 – Histogram indicating effectiveness at changing the swarm's largest principal axis length, L_{PA1}

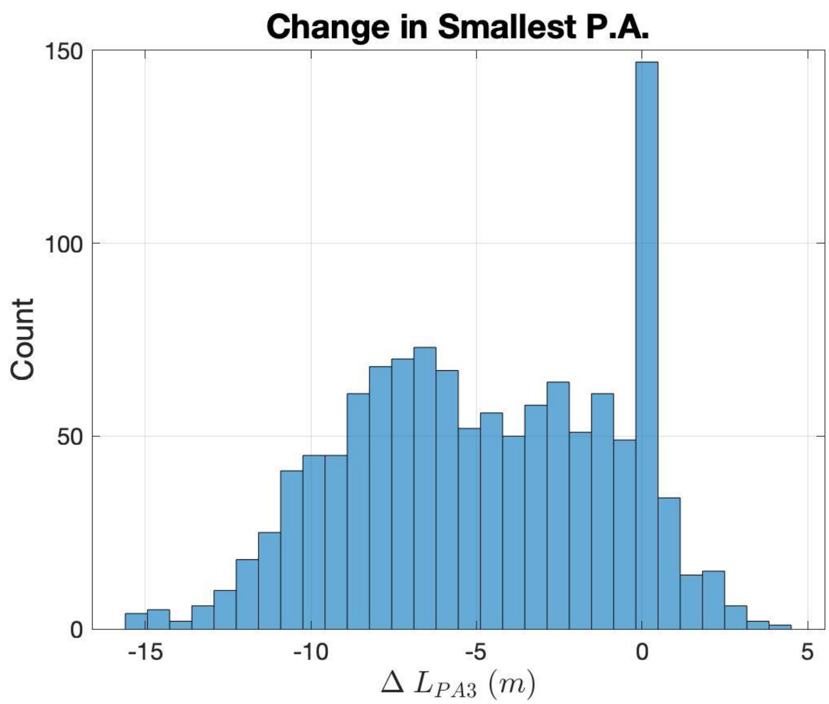


Figure 120 – Histogram indicating effectiveness at changing the swarm's smallest principal axis length, L_{PA3}

8.4 Comments on Negative Emergence and Assured Autonomy[338]

Another way to interpret the results of the swarm Yaw case is the manner in which it serves as ***negative emergence*** (undesirable emergence behaviors as discussed in [259]). For the "red team," the ability to force another swarm to maneuver off course is a success. However, for the "blue team" it is a serious failure, and an exploit. If the swarm were set up to perform a mission autonomously, blue team's decision-makers would likely test the swarm behavior in various non-adversarial contexts, and then deploy it in real-world missions (say, to survey a wildfire). Under the assumption of a non-adversarial context, blue team's decision makers might equip the drones to report back its performance using the standard performance metrics given in SwarmLab (safety, order, connectivity, etc.). In that case, the only metrics that might show unusual behavior would be order and safety. Suppose, however, that red team perfected the art of redirecting a swarm safely. The only metric remaining that blue team can use to know something is going wrong is the change in order. To blue team, changes in order would seem like the swarm is maneuvering around a very large, unforeseen obstacle, which they might dismiss as the fire spreading rapidly, or a large cloud of smoke, or a sensor malfunction. It might take tens of minutes, if not an hour, before the blue team decides to recall the swarm, which gives red team enough time to confiscate the swarm. Performing an exploitation analysis like the one described here, blue team would have noticed that most of the standard MoMs are insensitive to this maneuver, which would have prompted them to consider new MoMs and new requirements

[338] This author would like to thank Dr. Domerçant and Dr. Schrage for the insightful feedback that led to this section.

for their design (e.g. the capability of reporting and responding to deviations from their mission).

This leads to the question of how one can be assured that an autonomous system can perform a mission effectively and safely, as discussed in a work published just six months ago [260]. While assured autonomy is outside the scope of this thesis, it seems clear that the assurance literature is firmly anchored in SE and has not yet advanced beyond the gaps cited early in this thesis. The aforementioned document clearly echoes the problems with complex/emergent behaviors that inspired this thesis, such as:

1. "Assurance is context-dependent and not once and for all" [260].
2. There is a gap in current capabilities such as the "verification and validation techniques that enable the analytical evaluation of novel features such as non-determinism, complexity, and uncertainty" [260].
3. "The ability to trace the root cause of failures is critical in autonomy" [260]. Later in the document, "Because autonomous systems are typically heterogeneous" (as are most SoS, see references cited in Section 1.5) "their verification is likely to require a range of methods rather than a single unifying tool… The outcomes of these techniques should … help eliminate unintended functionality…" [260].

Compare those items to:

1. Kinsner's observation from 2010 cited in Section 1.6, "Complexity appears to be context sensitive, and cannot be defined universally, once and for all" [80].
2. Kalawsky's observation from 2013, also cited in Section 1.6 that "Development of reliable early detection of undesirable emergent behaviour… especially for Systems of

Systems" is one of the Grand Challenges in the *Verification, Validation and Assurance* of extremely complex systems [93].

3. The Research Problem of this thesis stated in Section 1.6, which was proposed to the committee nearly 18 months ago: "The traditional SE approaches to defining the properties and behaviors of a SoS that are distinct from those of its constituent systems lacks generality and traceability, and results in designs whose behaviors are only partially understood, the remainder of which can be exploited for some unintended purpose."

These statements by the community researching autonomy demonstrate that the research conducted in this thesis is still relevant, and that the problem of emergent behavior detection, and exploitation is still cutting-edge research. As illustrated by the adversarial boids and drone swarm case studies, the method in this thesis can be used to diagnose physically non-decomposable systems (self-organized systems), functionally non-decomposable systems,[339] and to identify opportunities for emergent behavior exploitation.

[339] More work is needed to obtain sufficient conditions for emergent behavior detection, but the necessary conditions compiled here are a good starting point.

CHAPTER 9. CONCLUSIONS AND FUTURE WORK

The development of a method for emergent behavior detection in SoS has been called one of the grand challenges in the verification, validation and assurance of complex systems [93], and is one of the major research areas for Systems Science identified by the International Council on Systems Engineering [94]. One quality that INCOSE has determined to be the opposite of an emergent behavior is a behavior that can be decomposed into smaller functions.

Since the early 2000's, the Department of Defense adopted a Capabilities-Based approach to its acquisition program, and since then has continuously conducted Capabilities Based Assessments and Fleet Synthesis Studies (see Section 1.3-1.4), both of which are designed to help guide the DoD's system acquisition decisions. The CBA and FSS both depend on the ability to decompose system requirements and capabilities into smaller sub-system functions, and sub-functions. However, most acquisitions involve complex systems or SoS that exhibit emergent behaviors, which are non-decomposable. The presence of unforeseen emergent behaviors, particularly undesirable ones, can make systems vulnerable to attacks, hacks, or other exploitation, or can cause delays in acquisition program schedules and cost overruns in order to mitigate them. As the DoD increases its acquisition of modular platforms, unmanned vehicles, drone swarms, and other advanced technology, the influence of emergent behaviors will only increase.

The research objective for this thesis is to develop a method for making quantifiable SoS emergent behaviors[340] traceable to the patterns of interaction of their constitutive systems, so that exploitable patterns identified during the early stages of design can be accounted for. This objective was chosen upon identifying several gaps in the SE literature:

1. Standard functional decomposition methods do not extend to emergent behaviors in the sense that independent functions at the extremes of the graph (those that can be *directly controlled*) no longer correspond to the coupled functions that *directly cause* the desired (emergent) behavior.
2. There is no method by which component interactions can be used to predict the existence of an emergent system-level property or behavior and *traceably* attribute a quantifiable, system-level property or behavior to that system.

In layman's terms, the problem of emergent behavior is like the problem of trying to reverse-engineer the properties of a molecule when the only information one has is the properties of a few isolated atoms. Unless one can determine how the atoms interact, how those interactions cause the atoms to arrange themselves, and how the resulting molecule interacts with other molecules, it is impossible to make quantifiable statements about the properties of molecules. The same analogy extends to any system exhibiting emergent behavior, but this thesis focuses on self-organized systems (e.g. flocks of birds, swarms of unmanned quadcopter drones, dogfighting pilots) in order to build a logical argument free

[340] Early in the text this problem is stated in terms of "non-decomposable" behavior. Those behaviors are simply referred to as emergent behaviors here for clarity and brevity. The definitions are discussed in Sections 1.5-1.7.

of circular reasoning. The literature review revealed that the two aforementioned gaps appear to be caused by a confluence of other gaps:

3. Some acceptable baseline set of definitions for emergent behavior is needed in order to build a useful ontology.
4. An ontology that accommodates emergent behavior and enables falsifiable claims of system "existence" is needed as a philosophical foundation for a mathematical method.
5. There currently exists no single mathematical method that performs all the steps needed to satisfy the research objective.

It is the lack of a mathematical method that prevents two final gaps from being filled:

6. There is no widely accepted approach for mining data to derive model of emergent behavior.
7. There is no widely accepted approach for qualitatively and quantitatively associating emergence with components.

Without this data mining and behavior association approach, there is no modeling or analysis method that will adequately characterize emergent behavior, and thus no decision-maker will adequately account for said behaviors in a systematic way. Currently, when engineers and decision-makers are faced with systems that exhibit emergent behaviors, they must rely on their personal education and experience to determine whether and how to exploit or avoid that emergent behavior. The current state of the art is often ad-hoc, combined with sets of best-practices disseminated via organizations such as INCOSE, or research methods limited in their scope. The main advantage of a systematic method, such as the one developed here, is that it can be improved incrementally (as opposed to reinventing the methods for each new problem), and its limitations are easier to diagnose and remedy because the underlying assumptions of the method are clearly stated and discussed.

9.1 Summary of Method and Research Findings

In order to derive a mathematical method for emergent behavior identification and exploitation, more information was needed about the two main objects of study: systems of systems and the emergent behaviors they exhibit. This led to the formulation of the first two research questions:

- Research Question 1: Which essential features of emergent behavior constitute necessary conditions that can be implemented in a mathematical/computational model?
- Research Question 2: What minimum set of data is necessary to simulate a SoS that satisfies the requirements for emergent behavior?

In order to answer those questions, it was first necessary to identify a set of useful, and minimally controversial definitions terms including, but not limited to: system, complex behavior, emergent behavior, and complexity. From these definitions, this research built an ontology from which it became possible to perform two important tasks (1) to identify a behavior in a falsifiable way, identify a system and its components in a falsifiable way, and to trace the emergent behavior to that system, and (2) to justify using numerical simulations as a test bed for studying emergent behavior. Thus, it was possible to address Gaps 3 and 4 from the literature, as well as parts of Gap 2. It should be noted that although Gap 1 motivates the research problem, the fact that emergent behaviors are non-decomposable in the sense presented here is an inescapable reality. Thus, the research objective (following Gap 2) is to create a method for addressing non-decomposable behaviors by making them traceable.[341] In the process of building the ontology, seven necessary conditions were found in the literature to be the answer to Research Question 1, and four pieces of required information were found, from the literature, to answer Research Question 2 (see CHAPTER 4).

The remaining gaps (Gaps 5-7 and the rest of Gap 2) were addressed by the development of a mathematical method. This process resulted in three additional research questions, each of which is accompanied by an experiment, because without experimentation it is impossible to determine that the predictions made by the mathematical method actually work in practice, or that the method itself is effective and useful. The three research questions were:

- Research Question 3: How many nontrivial quantitative emergent properties must a simulated SoS have?
- Research Question 4: Which quantitative properties are candidate emergent properties of a simulated SoS?
- Research Question 5: Once identified, how can emergent behaviors be exploited?

Thus the goals of these questions, and the method itself, was to determine how many emergent properties a system would have, how to identify them within a set of numerical data, and how to exploit the emergent property once it is found.

[341] This is a generalization of decomposability. Traceability enables the identification of cause-effect relationships where the cause is collection of coupled component interactions operating over time, rather than independent, sequentially executed functions (as it would be if the behavior were decomposable), and the effect is an emergent behavior (as opposed to a decomposable behavior).

It was found that Research Question 3 could not be answered as stated. Instead, Hypothesis 1 formulates an upper bound on the number of emergent properties a system can have based on the quantification of a form of data compression associated with the structure of the self-organized system. This quantity is calculated during the first major step of the method (the pattern recognition step). The tools for pattern recognition used in this thesis are found in the literature.

In order to answer Research Question 4, a set of numerical conditions were developed that combined the necessary conditions obtained for Research Questions 1 and 2, to a set of numerical criteria developed for this thesis. These criteria are part of the second major step of the method (the behavior association step). As the name suggests, this step associates a specific emergent behavior with a self-organized set of components (i.e. a system). By Hypothesis 2, the numerical criteria are considered sufficient conditions for emergent behavior detection.

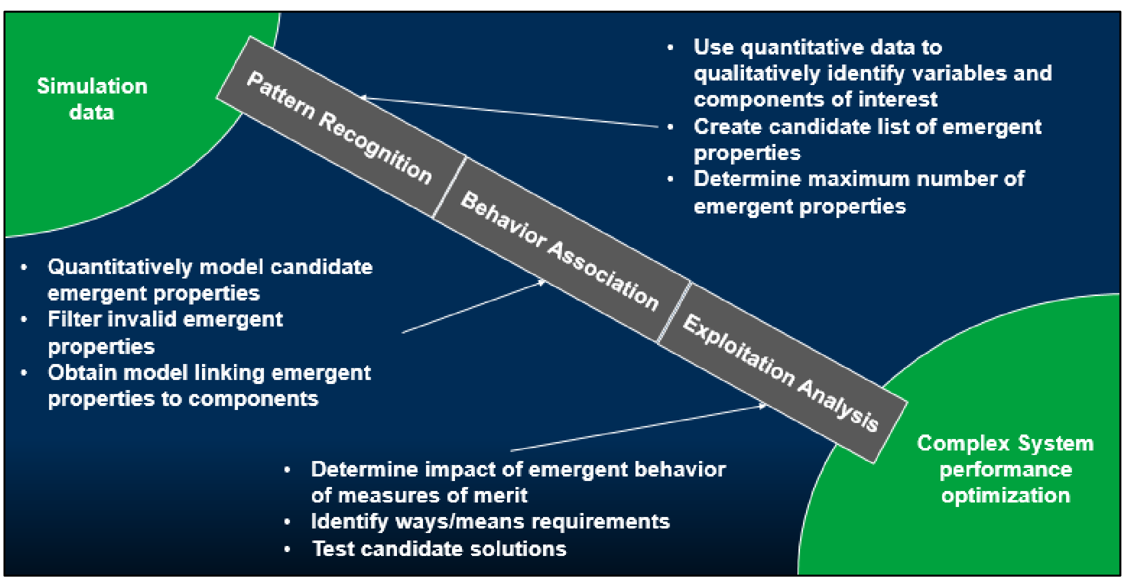


Figure 21 – Essential steps of Emergent Behavior Exploitation Method

Research Question 5 is answered using tools from the literature. It is determined that a simple sensitivity analysis provides ample information for ways vs. means decisions. When combined with simulations, the exploitation analysis, which is the third and final major step in the method, determines the impact of the exploit on system performance. This information can be used to develop system requirements, or identify a need for new measures of merit (if the existing metrics are insensitive to the exploit). The three major steps of the method are summarized in Figure 21, reproduced here for convenience. The intuitive nature of the sensitivity analysis does not guarantee the effectiveness of the exploitation analysis step. Therefore, to demonstrate that the exploitation analysis technique, and by extension the method as a whole, is effective, Hypothesis 3 states that targeting the system-level (emergent) property will be more effective than targeting the system components (component-level properties).[342]

Hypothesis 1 was falsified by experiment. The evidence suggests that either the maximum number of emergent behaviors is not a linear function of data compression, or that the maximum number of emergent behaviors is not determined at all by the self-organization of the components. In both CHAPTER 6 and CHAPTER 8, the number of possible emergent properties easily exceeded the maximum value predicted by Hypothesis 1. Upon examination of the arguments and findings in Dr. Vadim Kim's thesis [59], however, it seems that the latter explanation is probably the correct one.

Hypothesis 2 was also falsified by experiments in CHAPTER 6. Both a false positive was found, and the conditions were determined to be incomplete. Therefore, the

[342] This is phrased slightly differently in the main document because the components in question are simulated pilots.

numerical criteria are not sufficient conditions. However, the ability of the symbolic regression tools used in the behavior association step to distinguish data sets where system-level interactions had taken place from data sets where no interaction had taken place suggests that the numerical criteria can be added to the list of necessary conditions, and that overall, a robust list of necessary conditions for the identification of emergent behavior have been obtained.

Hypothesis 3 was supported by experiments in CHAPTER 7 and by example in CHAPTER 8. In both case studies, the sensitivity analysis yielded insights for emergent behavior exploitation, and the rules designed to target those behaviors had a dramatic impact on system performance. However, in CHAPTER 7 it was found that the exploit had a significant impact on the measure of merit, while in CHAPTER 8, the exploit had less of an effect on the measure of merits despite the fact that the exploit obviously succeeded and served a useful function. Thus, the results from CHAPTER 8 indicated that additional measures of merit were needed (particularly those related to performing specific mission tasks, rather than the more generic metrics common to drone swarm studies).

Another important consequence of Hypothesis 3 being supported is that the method was shown to work, and thereby satisfy the research objective, despite the lack of sufficient conditions for emergent behavior identification. Therefore, not only did the exploitation analysis step succeed in identifying an exploitable behavior, this behavior was also traced to the pattern of component interactions that generate the self-organized system (the trace is performed by the method during the first two steps). Therefore, this thesis succeeded in developing a method that achieves the research objective. Achieving this research objective addresses the research problem: the method in this thesis is generalizable, and enables the identification and tracing of unintended exploitable behaviors to the relevant components.

9.2 Future Work

The definitions, ontology, and mathematical method developed in this thesis provide numerous opportunities for future study. First, the pattern recognition tools used in this thesis rapidly become inefficient as the number of system components increases. Future studies can explore the use of a variety of sophisticated machine learning tools for pattern recognition in large data sets. Second, while numerous necessary conditions for emergent behavior identification have been found, future studies should continue to search for a set of sufficient conditions that can identify emergent behaviors in numerical time series data. Third, the sensitivity analysis used for the exploitation analysis step is a good baseline, but it does not prescribe a specific exploitation approach. Furthermore, more techniques for quantifying the performance of exploits, or for suggesting new measures of merit if the existing list is insufficient are needed.

Regarding the ontology in this thesis, the method described here satisfies the research objective for simulated self-organized systems and appears to be extendable to other system types and real-world experimental data. Nevertheless, future studies are needed wherein the method is extended to real-world systems and artificially-organized systems (i.e. designed systems) to determine whether there are additional challenges, or whether new tools are needed.

The method in this thesis is limited to properties that are represented by continuous numbers (as opposed to discrete or ordinal variables). Extensions of this method (or new methods) that deal with discrete periodic signals may also have to grapple with the well-known halting problem. This method is also limited to studies of simulations where the model is assumed to be valid and efficient. While those assumptions were acceptable for this work,[343] such assumptions can rarely be made for real systems. More research is needed to grapple with the challenges presented by imperfections in the simulation and discrepancies between simulated and empirically observed emergent behaviors. Furthermore, this method studies simulations with idealized components and idealized environments. More research is needed for simulations where the components are not idealized, as such simulations are typically needed for generating higher fidelity data. Finally, this thesis focused on systems with identical and interchangeable components. More studies are needed of systems with varied components.

It appears that emergent-behavior detection may be a fundamentally semi-empirical endeavor. It may be that emergent behavior is to systems engineering what turbulence is to fluid mechanics, and nonlinearity is to algebra. It is the set of infinite possibilities that follows after the simplest problems are well understood. Like turbulent flows, there will probably be some special classes of emergent behaviors that can be predicted and concisely modeled by observing that they conform to some simplifying constraint, while the overwhelming majority of them may remain areas where a human will have to exercise individual judgment.

Researchers wishing to follow in the philosophical framework of this thesis are encouraged to read the following:

[343] If the method does not work on the ideal case, it probably will not work on the real case.

- “General System Theory” by Hofkirchner and Schafranek [198]
- “Emergence: logical, functional and dynamical” by Mitchell [144]
- “Emergence Theories and Pragmatic Realism” by El-Hani and Pihlström [204]
- “An Information-Theoretic Primer on Complexity, Self-Organization, and Emergence” by Prokopenko et al. [206]
- “Challenged by Instability and Complexity ... Questioning Classic Stability Assumptions and Presuppositions in Scientific Methodology” by Schmidt [156]
- “On the Limits of Bottom-Up Computer Simulation: Towards a Nonlinear Modeling Culture” by Richardson [155]

These are not quite introductory texts, but systems engineers will have already received the necessary foundational instruction. Once again, the list of authors given in Section 1.7 is recommended: Baas, Ryan, Kubík, Wimsatt, Crutchfield, Abbott, and Minati. Finally, the reviews by Kim [59] and Jodoin [118] are thorough, but go in different directions.

Referring back to the “non-decomposable” nature of emergent behavior, it appears that a simpler analogy would be useful: self-organization is to physical decompositions what emergent behavior is to functional decompositions. Physical decompositions graphically depict components and their interactions. Some of those interactions take on the added significance of enabling a new arrangement of components. This new arrangement (if stable) becomes a system in its own right, and so should be depicted as a latent node in the original physical decomposition (see Appendix for additional discussion on latent node representations). Functional decompositions graphically depict actions (functions) undertaken by the objects at various levels of abstraction, and the relationships between lower-level actions and higher-level actions. Some of the relationships between actions take on the added significance of becoming a new process. This new process (if exploitable) becomes an action in its own right, and so should be depicted as a latent node in the original functional decomposition. This may prove to be a more fruitful contextualization of the problem going forward.

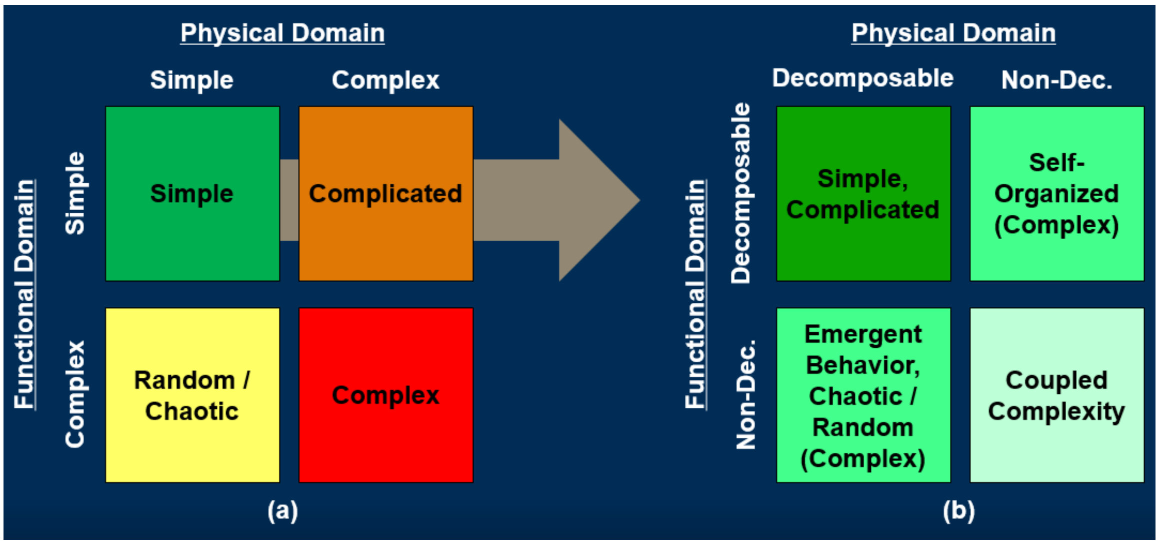


Figure 121 – (a) Complex system taxonomy adapted from [81], (b) modified taxonomy demarcated by decomposability rather than simplicity

As shown in Figure 121, it is possible to extend the complex system taxonomy first developed by Balestrini-Robinson [81], to a new categorization demarcated by the decomposable/non-decomposable dichotomy used in this thesis. Some slight changes to the classification of certain systems occurs. In this new taxonomy, systems that have coupled complexity would include most living biological systems, as well as many chemical systems. This taxonomy builds on the findings of this research, as well as the arguments by V. Kim [59] so that a system can be self-organized without exhibiting emergent behavior. In this taxonomy, systems that are physically complex but not functionally complex, or vice versa, no longer have a 1:1 mapping between component and function (see Stair Climber example in Section 1.6, and the latent node discussion in the Appendix). Other ways to cast the problem that may facilitate future research include:

- Emergent behaviors as the direct analogy of self-organization (as stated above): consider a set of functions arranged sequentially (decomposable), and compare that to a set of functions arranged simultaneously, arranged in parallel, or coupled together. Search for new functionality obtained as a consequence of the process in which the functions participate ("process" being the name for the arrangement of functions). In other words, consider the relationship between how coupled a system's functions are and how complex the resulting system is (in terms of new, system-level functionality).
- Emergent behaviors in association with data compression: consider a mapping that takes data compression measured in terms of information entropy and/or Kolmogorov complexity, rather than model time complexity, as an input and outputs the number of new functions obtainable as a result of the compression.
- Non-reductionist modeling: in keeping with Kitto's thesis, re-write the governing equations so that interactions and behaviors are the central "object" of the equation, as opposed to the physical components of the system. Then, just as patterns can be found in self-organized systems, patterns can be found in this new set of equations. That data compression could then map directly to the number of new behaviors/interactions since the pattern is based on a set of behaviors rather than a set of physical component properties.

Clearly there is much more work to be done on the subject of emergent behavior. While this thesis hopes to have made a useful contribution to the topic, it is only one of many more studies to come.

APPENDIX A. SUPPLEMENTAL MATERIAL

Topics briefly mentioned in the document are discussed here.

A.1 On Hypergraphs

Consider the Stair Climber example again, but this time represented using a hypergraph[344] wherein the higher level behavior exists because of the interactions of lower-level systems, but is not attributable to any a single or aggregated lower-level behavior [67].[345] In a typical physical decomposition (as in Figure 2-Figure 3 “phy” plots), one can represent entities (e.g. a system) as a node, and interactions between entities as edges connecting nodes. To create a hypergraph, a higher-level node can be made up of multiple, interacting, lower-level elements that, combined, enable the higher-level node to interact with some other entity via a behavior that is unique to the higher level node. That is, an edge connecting higher-level nodes can be created because a collection of interactions between lower-level nodes (contained entirely within a single higher-level node) exist that collectively enables the interaction[346] (see Figure 122). In this example, the Pull, Push, and Attach nodes are collected into the Lift node, and Move is depicted as interacting with Lift. The Move and Lift nodes are collected into the Climb Stairs node. Note that, unlike the hypergraphs in Figure 2-Figure 3, this graph does not have matching supply / demand nodes.

---

[344] A heterarchical structure could also be represented using a hypergraph, but would require permitting lower-level nodes to be members of multiple higher-level nodes and edges connecting nodes of different levels (a highly coupled hypergraph). An alternative would be a cyclical layered graph.

[345] Grisogono uses “emergence” in a sense nearly interchangeable with “complexity” as used here.

[346] Though not referred to as a hypergraph, an example of this can be found in Figure 2 of [284].

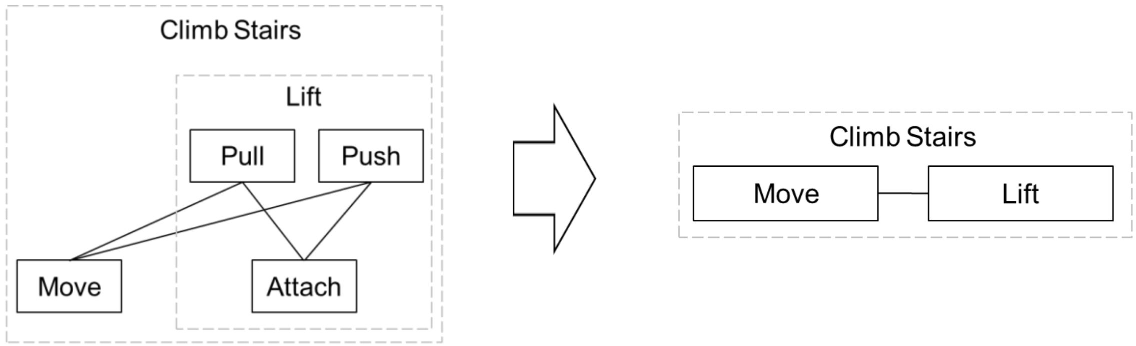


Figure 122 – Cyclical Layered Graph (Figure 8) redrawn as Hypergraph

That scheme was inspired by the "capabilities supplied / demanded" convention used in Grand Challenge work tangential to this thesis [261], and is intended for ease of use within a CBA/FSS context. In those figures, the hypergraph itself denotes causation: the nodes denote objects, the edges denote interaction between objects, and the collection of nodes with edges into a higher-level node denotes causation. If all lower level interactions occur at the right times, then the higher level nodes is caused to exist. Figure 7-Figure 122, on the other hand, draws from the convention used in [71] wherein the *edges denote causation*. Regardless of the convention used, it is clear in Figure 122 that the Lift function is a node of nodes, and that Move must cause Lift in order to create the Climb Stairs function. This convention is revisited in Section 2.3.

A.2 Additional Arguments on Simulated SoS

The following comments are extensions of the discussion in Section 2.2.2, and are not necessarily sequential.

Once the simulation is expanded to include a SoS, the simulated SoS necessarily possesses the same *types* of uncertainty as a real SoS. There are, however, two *sources* of epistemic uncertainty that a simulated SoS *cannot* possess. The first is the epistemic uncertainty caused by the discrepancies between the simulation and reality. In a real-world experiment, the simulation never perfectly captures reality. The second cause of epistemic uncertainty is the engineer's limited knowledge of the environment. Both issues has been eliminated by Assumption 1. More importantly, limited knowledge of the environment can be eliminated using the fifth idealization.

Clearly, it is within the power of the programmer to eliminate component-related idealizations. For example, (1) a programmer uses procedural generation to define component properties and sub-components *ad nauseum*, which eliminates simplicity and indivisibility, (2) components can be programmed to delete under certain conditions, eliminating persistence, and (3) machine learning techniques are so sophisticated that their resulting behaviors routinely surprise their programmers,[347] thereby eliminating predictability. The key concept is that when a programmer does implement these idealizations, the system cannot automatically inherit them (just as real systems built by engineers cannot inherit them). In simulations, a system is always divisible. If the system exhibits complex behavior, it is no longer predictable, and even when it does not, its cause-effect relationships may be contingent on multiple, simultaneous component interactions, which eliminates any clean, one-to-one cause-effect mapping. Finally, whether the system is persistent is no longer trivial. Persistence in a system context typically implies some form of stable operation. Interactions among components or with the environment can disable the system, and so, in a functional sense, the system may no longer persist.[348]

---

[347] For example [180] discusses a simulated game of hide and go seek, where the ABM agents develop "surprising" strategies. Thanks to James Pagan for suggesting http://openai.com/blog/emergent-tool-use/.

[348] This is where the campfire versus pile-of-sticks contrast becomes meaningful. For example, is a dead battery still a battery? For nominal purposes, yes, but functionally, no it is not. In informal military jargon, this also refers to the concept of an operational kill, which is outside the scope of this thesis.

The theoretical maximum number of levels of abstract interaction between simulated components, systems, and SoS, however, are a function of all possible combinations of components (without replacement), and all possible combinations of systems (without replacement) with the rule that every system must contain at least two components, and every SoS must contain at least two objects (system/component combinations being permitted).[349] This massive number[350] says nothing of the number of properties that exist at each level (some levels could be empty[351]). Finally, there is the assumption that components only operate within a single environment. If the mind is abstractly represented as an environment unto itself (an abstraction permitted by the observation that humans can respond to ego threats and physical threats similarly [262] [263]), then the argument can be made that intelligent agents operate in two environments simultaneously (one perceived, one real), that are coupled together via the body. Such considerations fall outside the scope of this work.

It is worth noting that the challenges of emergent behavior present in SoSE also exist within a simulation. First, SoS boundaries (in the SE sense) are not always obvious because patterns of interaction can shift over time leading to different configurations of systems and/or components, as discussed in [183].[352] Within a SoS simulation, it is possible for the definition of a particular SoS to become meaningless before the simulation terminates (i.e. the systems or interactions no longer persist, or their interactions change).

[349] This is based on the assumption that any pairwise, ternary, etc., interaction of components can become a system unto itself (leading to pairwise, ternary, etc., interactions among systems, becoming a SoS). This generalizes to SoSoS…

[350] Szabo and Teo also remark on how rapidly the number of candidate properties of a complex system can grow from just a few basic components [96]. See also Kim's discussion of bottom-up simulation [58].

[351] Again, there is no one-to-one correspondence between physical and functional decompositions.

[352] Note also that they use weighted, labeled multigraphs (not hypergraphs).

This raises the question of whether the new configuration has become a different SoS with properties unrelated to the previous configuration. It may not be possible to answer this question in a simulation of fixed scale (much like it cannot be answered in a real experiment with limited observations). Second, since systems do not inherit the idealizations of their components, partial or collective system behaviors/interactions are not necessarily attributable to the SoS. For example, systems that are assigned to a SoS may not have operational envelopes that overlap.[353] If only a subset of the systems participate in an activity, is that truly a behavior of the SoS? It certainly cannot be an emergent behavior of the SoS, since it is inherited directly from its constitutive systems (and falling outside the scope of this study). Similarly, the failure of a real SoS to behave as expected may require component-level or system-level redesign rather than SoS-level redesign. Thirdly, consider cases with systems that possess partially or completely redundant behaviors (interchangeable systems). If systems are physically interchangeable but functionally equivalent,[354] which can happen with ships in a fleet or subsystems on a modular ship, then the systems are indistinguishable by their functional decompositions. However, the seemingly minor differences between systems can produce wildly different emergent behaviors at the SoS level. Since the purpose of a functional decomposition is to distinguish objects via their behaviors, this approach creates a paradox wherein two SoS are only distinguishable by their emergent behaviors, which can be very difficult to

[353] Suppose a SoS is made of 5 systems. Sometimes 3 systems will operate together, sometimes all 5, sometimes all 5 will operate independently, sometimes 2 systems will cause 1 system to fail, etc.
[354] This forms part of the basis of evolutionary SoS.

detect/predict.[355] This same issue occurs in practice. Thus, many of the challenges of SoSE remain present in SoS simulations.

A.3 Equation-Free Modeling

Beginning in 2000, Equation-Free Modeling (also called Equation-Free Multiscale Computation) has been proposed as an alternative method to emergent property or behavior identification (termed macroscopic property/behavior) using component-level simulation data (termed microscopic) [264] [265]. Essentially, this technique creates two samples from a single data set of simulated low-level components. One sample is taken over very short time steps, and a second "coarse grain" sample taken over much larger time steps. By comparing the dynamics of both samples, this method attempts to extract higher-level property information from low-level data by filtering out long-term trends that are distinct from short-term behaviors without ever generating an explicit equation that relates the two sets of properties. Thus, the method relies on three assumptions: (1) a model[356] exists that relates the microscopic properties to macroscopic properties, i.e. upward causation is occurring, (2) a macroscopic property exists and, in general, is known in advance, (3) the proper length and time scales are known in advance, which dictates the time steps for sampling the data. In many applications, researchers derive the macroscopic properties from experiment, and so the time scales are known in advance (i.e. the duration of simulation needed for some macroscopic property to appear can be estimated in advance) [195] [265]. In this case, the main challenge left to the researcher is to select the time step size. The specific steps are more involved, and present additional challenges, but this

[355] Thus evolutionary SoS taxonomies can fall victim to the fallacy that a velociraptor is an archaeopteryx which is also a pigeon depending on the level of detail of their functional decompositions.
[356] Not a simulation, which would be trivial, but a formal mathematical model of some kind.

characterization suffices to distinguish the capabilities of the equation-free method from the scope of work in this thesis at a philosophical level.

First, this thesis will not assume that a model exists relating high-level behavior with low-level behavior *throughout* a simulation because, in general, this assumption does not hold with regards to emergent behavior. The ultimate reason for this, as argued in Section 2.2.2, is that systems and SoS, when simulated, are not persistent, indivisible, or predictable. Since the macroscopic property is the property being sought, if the SoS possessing that property does not persist, then the property does not persist, therefore, the model becomes undetectable once the SoS vanishes. If the object is divisible (while retaining its definition, as flocks are commonly thought to be), then there can be multiple simultaneously occurring instances of a system, which will confound measurements and complication the process of finding a model. Finally, the need for predictability goes without saying. For example, in a simulation that runs for a long period of time, it is very possible that a macroscopic property will appear in the data at multiple distinct points in time, and last for different intervals of time, and that the majority of the simulation time will correspond to the formation, dissolution, and non-existence of that property.[357]

Second, imagine that a molecular dynamics simulation could be executed for a drop of water left outside on the pavement during winter, and the simulation generated data representing 24 hours of behavior. Over the course of that day, under the right weather conditions, that drop of water can grow (due to precipitation), shrink (due to evaporation), or freeze (due to the cold). If the desired macroscopic property is the Young's Modulus of ice, it makes no sense to sample the data when the water is in liquid or gaseous phase (lack

[357] The analogous argument to this research can be made by replacing the terms "macroscopic property" with "emergent behavior" or "property of higher-level object" or SoS, etc. In other words, if the SoS breaks during a simulation, then whatever model of high-level behavior there may have been becomes invalid for the rest of the simulation. So, while there may have been such a model, it does not make sense to assume one can take meaningful measurements of the properties in question throughout the simulation. Once the SoS breaks, the model breaks too.

of persistence). Furthermore, the Young's Modulus becomes difficult to measure when the drop is experiencing a phase change or when snow or rain comes into contact with the frozen ice on the ground (lack of indivisibility, and predictability). Only when the water is frozen, and left in relative isolation, can one say safely assume that "a model exists" that relates microscopic to macroscopic properties.[358] This a major hurdle within unstable, non-equilibrium simulations, and illustrates why the second assumption (that a macroscopic property exists and is known in advance) cannot be expected to hold for emergence.

Third, in general, there are no grounds for selecting any particular time step or simulation duration *a priori*, particularly when the macroscopic behavior in question is unstable and nonlinear.[359] This helps explain why researchers that utilize experimental observations have had success using equation-free modeling [264] [195], but also why prior knowledge of macroscopic properties is not a sufficient condition for the equation-free method to succeed [265]. When Samaey, Holvoet, and De Wolf applied the equation-free approach to simulated data exhibiting self-organization[360] to discover unknown macroscopic properties, they were able to successfully identify "aggregative properties" (in this case, expected values and standard deviations) [195].[361] By the definitions in this thesis, Samaey et al. identified non-emergent properties of a self-organizing collective. While their work was a success by the goals of their research, and other researchers have introduced additional analysis tools to improve the method's efficacy, the authors nevertheless warn, "One conclusion is that the application of equation-free analysis is by far a non-trivial exercise. The iterative application of the technique however proves helpful

[358] This assumes a stable model (within the context of patterns and self-organization). A terribly nonlinear model may also exist, but unambiguously identifying that model would require that only two levels of abstraction exist throughout the simulation, which is not guaranteed.
[359] Self-organized criticality [194] [291] shows that slowly organized phenomena can cause sudden and severe changes in an environment (e.g. earthquakes).
[360] Since the macroscopic properties were not known in advance, he dubbed this an exploration for emergent properties.
[361] While emergent properties are nonlinear, a high level object can certainly have linear properties.

in thoroughly understanding the link between microscopic and macroscopic behaviour" [195]. While equation-free modeling is a promising approach, the form of analysis it is designed to facilitate is outside the scope of this thesis.

A.4 Periodicity Revisited

In the case of periodic behavior, however, the behavior itself provides two key pieces of information that can be used to derive an appropriate time step size and simulation length: they are the period and amplitude of the periodic function. To the extent that emergent behaviors can be traced to patterns, there is no reason to believe that an emergent behavior can be properly observed over a time scale smaller than one period of oscillation. To empirically confirm that an emergent behavior is taking place, at least two periods are required (otherwise, there is no way to confirm that the first oscillation was periodic), which determines a lower-bound on the required simulation time. Furthermore, the extent to which the pattern measurably manifests as a macroscopic property at some larger time scale is influenced by the amplitude of the oscillation. Suppose the periodic behavior is a mechanical vibration (like a weight on an undamped spring). The amplitude of oscillation becomes a minimum length scale as well as a maximum perturbation. The manner in which that perturbation propagates throughout the system depends on a variety of factors, but under the right circumstances, the effect can be damped out so that, upon "zooming out" far enough, the overall behavior appears stable. This can be treated using order of magnitude analysis techniques such as those common in [160], and in this way, periodic behavior can provide a mathematically rigorous basis for associating resolution with scope and scale, as discussed in Section 1.7 In short, following the example provided, there are two intervals in question here, and the largest of the two is the smallest interval needed to observe emergence: (1) time: twice the period of oscillation; space: greater than the length of the oscillation, or (2) time: the time between stabilized configurations of self-organized objects, whose pattern was perturbed by some outside entity; space: the length/area/volume interval over which the self-organized object transitions from one stable configuration to the next in addition to the space needed to observe the object causing the perturbation. Although Ryan does not make this exact mathematical case, he similarly argues, "[a novel emergent property] arises from structure that is extended over the scope of the system… There is a difference between local and global structure in any system that exhibits emergent novelty. This explains why emergent novelty cannot be understood or predicted by an observer whose scope is limited to only one component of a system" [111]. To the extent that metastable pattern compression is lossy, this also means that the impact of information loss may scale, or, at the very least, that it will impede the observation of the cause of some emergent behaviors at higher levels. In the case of chaotic behaviors, which are inescapably lossy, it is still possible to apply the aforementioned scaling techniques, but scales would be based on the bounds of the chaotic behavior, which may be visited very rarely compared to periodic oscillations.

One more implication of periodicity is that if the oscillation repeats for a time interval orders of magnitude larger than the period, then the probability density function of the amplitude will become sharply multimodal (the simplest case is the so-called inverted Bell curve observed for $y = \sin x$), which can be used as the basis for simplifying assumptions. Although these features will not be used on the canonical study at hand (because the periodic function is the constant function), they provide important information for time scale selection.

A.5 A Canonical Example of Emergence

This section is provided for completeness, however, this example is too far removed from the intended application of this thesis to serve as a numerical experiment. The narrative from the previous section is resumed in CHAPTER 4. Conway's Game of Life (GoL) is a zero-player game where a user can instantiate a white grid with black squares (thereby switching them “on”), and then run the game to observe what other grid squares switch on or off as a result of the GoL rules [266] [267] [268]. A sample set of rules are given in Figure 123.

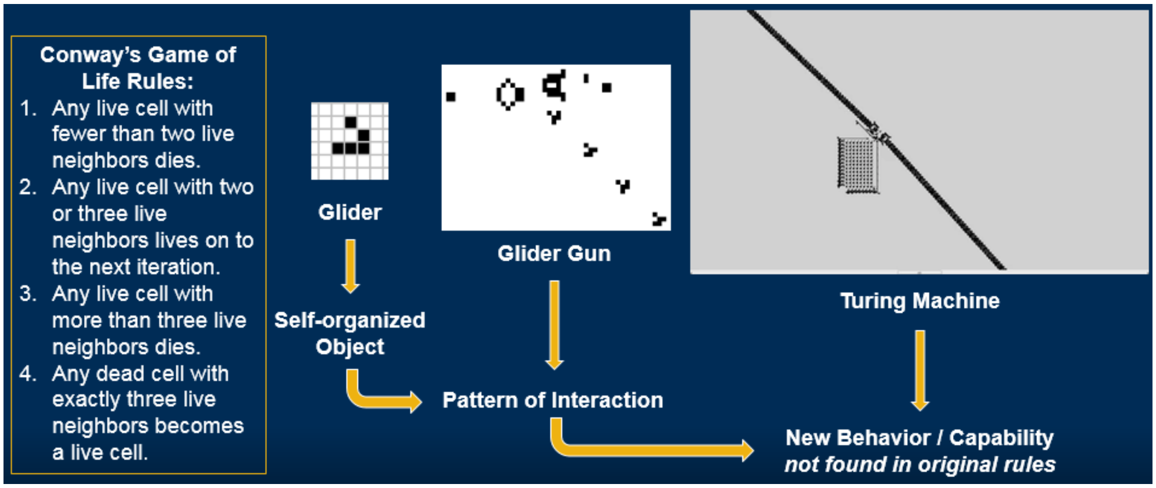


Figure 123 – Conway’s Game of Life Progression of Self-Organization – images and rules excerpted from [266] [267] [268]

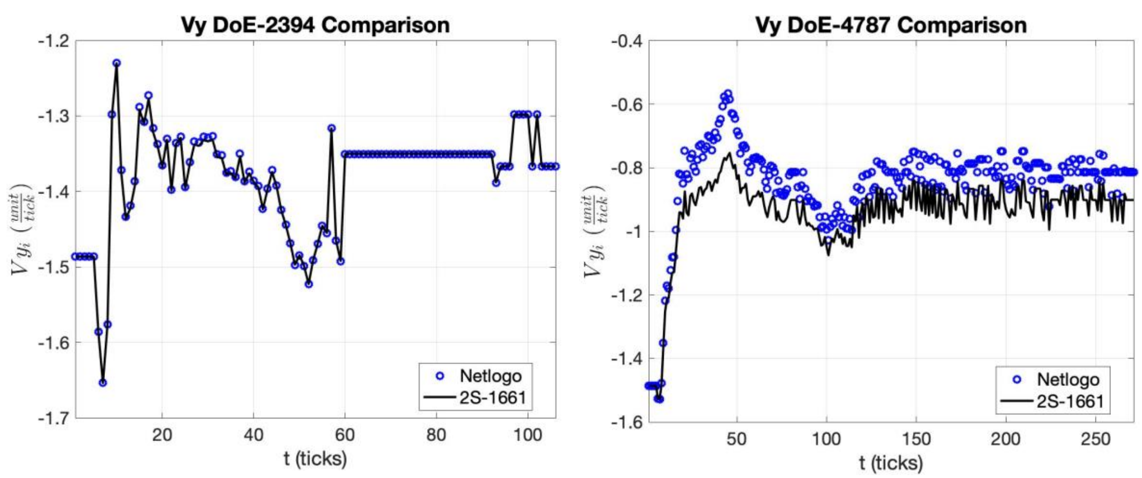
Users of the software have observed that some initial configurations will result in self-perpetuating patterns. In fact, some configurations will appear to move across the screen indefinitely (such as the Glider shown in [266]). Clever users later noticed that certain initial patterns could be distributed in space such that the patterns they generate interact to form new infinite patterns (commonly known as Gosper’s Glider Gun shown in [266], and in Figure 123 simply as “Glider Gun”). Over the course of several years, more researchers were drawn to build increasingly complicated mechanisms [269] until, in 2010, Paul Rendell created a full-blown Turing Machine inside of the GoL [267] [268], which is to say: a computer in a game in a computer. On their surface, nothing about the GoL rules indicates that a programmer can instantiate a Turing Machine. It took a substantial amount of time, curiosity, and human ingenuity to observe the patterns, identify their geometric /

organizational properties (e.g. center of gravity and velocity), and then space them out in such a way as to make other patterns that can interact to perform more complicated functions beyond merely self-perpetuating and moving.[362]

Finally, it is worth noting that this example illustrates emergence by design, which is not a topic explored in this thesis. This thesis relies on self-organization so that the criteria and methods given in CHAPTER 4-CHAPTER 7 can be tested objectively.

A.6 Time-series using SISSO

Although SISSO was not designed with time-series data in mind, there is a very straightforward workaround: simply replace the labels in the "materials" column with time stamp labels such as "ts5" for time step 5).



(a) (b)

Figure 124 – Time series of 2D SISSO descriptor trained on DoE #1661 and extrapolated to (a) NetLogo DoE #2394, and (b) NetLogo DoE #4787

[362] It is a near impossibility to derive the Turing Machine from a random initialization. Once again, we see that emergence is highly sensitive to initial conditions.

If the user wishes to have SISSO treat time as an independent variable, an extra column would have to be added to the train.dat file with the appropriate time step values for each row of data. Figure 124 depicts a model[363] produced by SISSO that was trained (regressed) on the time series corresponding to simulation #1661 (out of 5,000 simulations of flock interactions in the DoE), and then used to predict the time evolution of the same flock-level property in simulation #2394 and #4787 (i.e. extrapolation). In Figure 124a, the model extrapolates very well, fitting the Netlogo time series indicated by blue circles. In Figure 124b, the model reproduces the trend of the actual simulation in a qualitative sense, but consistently underestimates the true value after roughly 10 time steps.

Unlike Approach 1, Approach 2 relies on path-dependent models. The value of the property at the next step in time is related to the value at the previous step. In Figure 124, the data was modified such that the initial condition was subtracted out from every point in time (the model of "$V_y(t)$" was fitted to the data of "$V_y(t) – V_y(t=0)$" and then plotted by reintroducing the initial condition of the simulation). There are many ways to obtain such a model, and in the field of Econometrics, it is common to use a special recursive function called an auto-regressive model (AR model).[364] Many well-known functions have closed-form AR models.[365] For example,

$$\left.\begin{array}{c} Y(t) = \sin(\pi t/10) \\ Y(t) = 2\cos(\pi/10)\,Y(t-1) - Y(t-2) \end{array}\right] [270] \qquad (76)$$

[363] The term "2D descriptor" means that the model is a linear combination of two terms (e.g. $y = 3e^x + 5.2sin(x)/x + 0.12$). Usually those two terms are nonlinear.
[364] The models in Figure 124 are not AR models.
[365] The interested reader is referred to [356] [356] [356] for accessible introductions to AR models.

$$\left.\begin{array}{r}Y(t) = 1.1^{t} \\ Y(t) = 1.1Y(t-1), \qquad t > 0\end{array}\right] \tag{77}$$

AR models such as Eq. (77) are an example of auto-correlated functions. When studying auto-correlated functions in search of causal mechanisms (property x directly "causes" property y to change) a number of statistical issues arise beyond the problems associated with common goodness of fit measurements. For example, just as the function is auto-correlated, the error of a regression on the data can be auto-correlated, which will bias the results and typically requires a correlogram to identify. Furthermore, although a variety of tools exist for studying AR models, many of the approaches that use these tools assume that major trends in the data are not explanatory in a useful way (such as seasonality in data [271]), or that the underlying variables are random [272], neither of which apply to this thesis. Also, note that the AR models in Eq. (76)-(77) are exclusively functions of time. This form essentially strips all obvious causal information from the model, so that now each time step is predicted based solely on the data of the previous time steps and the coefficients of the equation. In many real-world problems (e.g. climate change studies), this is an inescapable limitation of the data. In this thesis, it is tantamount to removing the causal information in order to apply statistical techniques that hopefully recover the same causal information. Finally, this thesis is tasked with the challenge of replacing a valid, obvious causal model (models of upward causation), with an elusive causal model that requires justification (models of emergent behavior). It is unclear that AR models and their associated tools will be any more or less effective than regressions in obtaining or justifying the emergent behavior model. Therefore, after some investigation, AR models and their associated tools have been excluded from this thesis.

A.7 Property Down-Selection Considerations

Once a set of properties have been generated and mapped to a set of inputs using SISSO, candidate property equations that are inaccurate (per the statistics in Section 5.1.1) or computationally inefficient can be filtered out. However, the remaining list may still be long. Although the experiments for this thesis are based on the aforementioned metrics, it is worthwhile to discuss three more qualitative features that have received little attention: dimensions (as in, units of measurement), saliency, and sensitivity to perturbation (of the property, not the self-organized structure).

In terms of units, an argument can be made that the units of the candidate property should be consistent with the units of the properties of the self-organized group (e.g. the length of the flock is the sum of the inter-boid displacements, therefore one derives its units from the other). However, many equations in physics have coefficients with units that resolve the incompatibility between the units of the output and the units of the inputs (for example, Newton's equation for gravity has the constant G). Therefore, there is no guarantee that the units of the two entities will correspond or have some kind of linear relationship (e.g. the intermolecular distance of water crystals largely determines the volume of ice). Nevertheless, in the absence of such correspondence, one might wish for a compelling reason to accept the statistical correlation between two quantities. Finally, note that SISSO does account for units when generating algebraic combinations of terms[366].

A stronger quality, of course, would be saliency. The more often a particular property appears in various interactions, the better the chances that it is meaningful. Another way of

[366] At least two symbolic regression tools emphasize this feature in their documentation. It is likely a standard expectation at this point. See Appendix.

defining saliency would be to run the simulation under multiple initial conditions to ensure that the relationship between the high-level property and the interaction is reliably predictive. Doing so does not serve the purpose of testing hypotheses in this thesis, and so it will not be used. Nevertheless, studies of cause and effect[367] (e.g. model discovery studies) would do well to consider saliency as an important constraint.

This leads to the third quality, which is sensitivity. It is not unreasonable to expect that higher level properties are insensitive to lower-level fluctuations (in general), but this is not always the case. Cases where the higher level property may be vary dramatically can result in a form of instability at the higher level of organization. That may lead to disruption of the high-level self-organization, or the instability be the driver for the next form of emergence. For example, one might consider micro-fractures in a metal beam to give rise to plastic deformation. Destruction of low-level self-organization gives rise to permanent changes in the self-organization of the beam, which is then reflected in the value of its high-level property (the Young’s Modulus). Too much stress, however, and the part will break. On the other hand, in animal muscle tissue, micro-fractures signal the body to heal and strengthen the tissue, leading to increased overall strength, which can result in a change to the behaviors that the organism is capable of exhibiting (e.g. gymnastics, martial arts).

A.8 Brief Review of Time-series Analysis Using Granger Causality

If a collection of time series are represented as AR models, it is possible to use Granger Causality (GC) to determine if the trends occurring in one time series are likely to consistently foreshadow similar trends in another time series. Despite the name, GC is not exactly a test for causation (see [273] and the Appendices), and it cannot be used as a stand-

[367] “Cause and effect” are used here in the engineering sense of the term (e.g. a force “causes” a mass to accelerate). This is an example of terminology that causes confusion between philosophers and scientists. Another example discussed Section 1.7 regarding the “existence” of a “physical object.”

alone test. Following the procedure outlined by Econometrician Dr. Dave Giles [274], testing for GC must be performed in conjunction with two tests for stationarity (one to test against false positives, and the other against false negatives), and a test for co-integration in order to verify its results (co-integration in a time series implies Granger causation, but the converse does not hold).[368] This requirement presents numerous difficulties for this thesis. First, standard co-integration tests require estimating the order of integration of the time-series by repeatedly differencing the data until stationarity is achieved [275] [276]. Differencing will only render a time series stationary if the underlying trend is a finite polynomial. While it is true that many functions can be approximated over finite time intervals using truncated polynomials, this is extremely problematic because finite polynomial interpolations are notoriously bad at extrapolation, which means that the residuals are almost guaranteed to contain biased estimates due to an incorrect mean trend. Furthermore, this thesis strives to make statements about existence, not merely regressions on the data. Therefore, the estimated trend must be as accurate as possible (nonlinearity cannot be swept under the rug). Second, more powerful methods have appeared in the literature for treating time-series with non-polynomial trends using multiple approaches [277] including Fourier series [278]. While this would certainly better capture nonlinear trends, it raises the question of what it means to compute an order of integration since the Fourier series can be extended to fit nearly any data set. Furthermore, connections between this approach and GC tests are not readily available in the literature (as would be needed for a non-expert practitioner such as this author). More importantly, however, is the fact

[368] Dr. Giles recommends using the Johansen test. An alternative is the Engle-Granger test. The Matlab Econometrics toolbox [368] implements these tests (cointegration, stationarity, and GC) as independent functions. Some expertise is required to implement Dr. Giles' procedure, and it is unclear that an arbitrarily designed AR models can be supplied to the Matlab functions.

that a FS trend is only meaningful if the underlying trend is truly periodic, which need not be the case with regards to emergent behaviors.[369] Finally, the biggest drawback of GC (despite its association with emergence in parts of the literature [219] [279]) is that the time series of the causal variable and the time series of the effect variable must exhibit a significant lag in time between correlated trends. If the causation is genuinely instantaneous[370] (as it is for the cases studied here[371]), GC does not apply. Therefore, after careful consideration, it has been excluded from this thesis.

As stated in Section 5.1.1, there is one line of research that merges data compression measures with AR models to test for causality [220]. Although the authors show their results are superior to the standard Granger causality test (and Transfer Entropy calculations, as in [280]), and the compression framework has parallels with the discussions in CHAPTER 3, their work will not be utilized in this thesis since AR models are not being used.

A.9 Reasons for No Emergence or Surprise (in Simulation)

There are a number of reasons why an emergent behavior may never materialize or defy the expectations of an analyst. In the case of no emergence:

- If the components in a simulation all self-organize into one massive system so that there is nothing left in the simulation with which to interact, emergence will be undetectable (this limits the scale of properties that a simulation can predict).

[369] The periodic nonlinearities are found in the self-organized structure, not necessarily the emergent property (the equation based on upward causation) or the equation describing the interactions that involve emergent properties (the equations relating high-level properties to each other).
[370] Some data sets may appear to have instantaneous causes due to insufficient sampling [365].
[371] As an ABM expands in scope (literally including more agents and more space for them to interact in) the timescales will increase, including the time required for a cause to have a recognizable effect.

- The external components and system have no means by which to observe and act on the emergent properties of the new system (e.g. a chameleon changing color means nothing to a color-blind predator).
- There are too few components, oversimplified components, or an oversimplified environment, and so no self-organization occurs.
- The behavior is rare and the design space has not been sufficiently explored.
- The time-scale over which the behavior makes a noticeable impact on the system is significantly longer the time interval over which the data is obtained.

In the case of surprising emergence (simulated emergence inconsistent with experiment):

- Incorrect rules for component behavior.
- Numerical approximation in simulations of highly unstable nonlinear models.
- Errors in the code.

If the necessary interaction exists in the simulation, however, data mining tools and modern computing power provide the ability to search the space of simulated properties efficiently for good candidates. Once a set is found, the association between the property of a system and property of an entity external to that system can be made

A.10 Comments on Causation and Graphs

Graphs are very powerful mathematical representations. Just as they can be used to model physical and functional decompositions of systems, and have been successfully used to map system decompositions and interactions to computable models that quantitatively predict system behavior (system dynamics is precisely that [281] [282]), they have more recently been honed into specialized methods for extracting causal information from numerical data. Fantastic, accessible introductions are available in [283] [284]. Dr. Judea

Pearl's work has gained wide recognition recently [285]. The text by Kline [286] provides a much broader introduction to the field (including Pearl's work). These causal graphs are powerful to the extent that they can inform and guide experimentation, but beyond that, they are a purely mathematical approach to knowledge discovery that will face the same fundamental limitations any mathematical approach will face. Systems engineers should note that much of the work on these causal graphs has been done in fields that cannot perform experiments in the philosophical tradition of Karl Popper (e.g. social sciences, econometrics, etc.). The challenge faced by those fields is that they can rarely change the variables they are studying (see Chapter 17 of [284]).[372] That challenge extends to emergent behaviors, but to a lesser degree: the variables cannot be directly controlled,[373] but they can be systematically influenced via lower-level functions, and the resulting trends can be analyzed.

As suggested by Sections 1.6-1.7, the inspiration for this thesis was the realization that standard decomposition graphs cannot represent emergent behavior without a contradiction in notation, or a form of indeterminacy. If an edge represents the interaction between two objects (as in physical decompositions, where *the nodes are the objects*), then it cannot also represent the membership of some nodes within a higher-level node. If *a node represents a function*, and edges represent one function causing another (as in functional decompositions), then when multiple, coupled functions also collectively cause a higher-level function, it is impossible to determine from the diagram which function

[372] This extends to military planning, as discussed in the first sections of CHAPTER 2.

[373] For two fixed levels of abstraction (the system and its components) emergent behaviors can exist as latent variables within a complex system (either unforeseen high-level functions, or unforeseen intermediate behaviors/functions that affect known high-level functions). Corollary: the representation as latent variables (like the hypergraph) attests to the "objective unpredictability" of emergence. Even after being discovered, it is still an intermediate node (or a higher-level node in a hypergraph, etc.). See Figure 126.

comes first, or how many functions happen simultaneously, or how long the coupled process must occur before giving rise to the higher level behavior, etc. (indeterminacy). This is the sense in which emergent behaviors are not decomposable. The challenge worked through in this thesis was the challenge of formulating the approach for overcoming this non-decomposability using the knowledge obtained by academic disciplines that are all asking different questions. Philosophy asks "How do we define what is real, and understand real causation?" Engineering and medicine ask "How do we best interact with reality to influence its outcome?" This is a subtle but significant shift in focus, because it reveals that engineers make many assumptions about the nature and existence of reality. The social sciences can be said to ask similar questions to those posited by engineers and doctors, but often cannot validate their answers with direct experimentation. Thus, another challenge for this thesis was to collect the right answers to the right questions, and construct a method for data mining and experimentation using the right tools (summarized in Figure 21).

Before proceeding to a variety of important tools that can be used to build on the work in this thesis, the philosophical question of how causation is characterized within the context of mathematical models must be addressed. Although there are many avenues for stating the problem and ways to address it (see [287]) the simplest version with regards to emergent behavior is perhaps Gödel's Incompleteness Theorem.[374] In short, GIT was a response to an old aspiration: If all quantifiable knowledge in the universe can be expressed using mathematical concepts whose validity can be proven using the appropriate set of assumptions (axioms), then the discovery of all knowledge is a simple matter of finding the complete set of axioms and deriving all knowledge from those axioms. Gödel proved

[374] See also Kitto's review of the "Foundations of Mathematics" [73].

this to be impossible. According to his theorem, mathematics is faced with one of two alternatives: either the list of axioms is complete and contains a contradiction (thus, anything can be proven, including fallacies), or the list of axioms is incomplete and there exists some true statement that cannot be deduced from the axioms. Therefore, there will never be such a thing as a complete, contradiction-free mathematical modeling of all knowledge. This includes any mathematical construct that aspires to be causal. There will always be the potential for confirmation bias, and all of the usual misleading fallacies in any approach. That is not to say causal inference schemes are futile, of course. It simply means that "causal inference cannot be reduced to a collection of recipes for data analysis" [283]. To repurpose the phrase by Dr. Laughlin, emergent behavior discovery requires "constant hand-shaking between theory and experiment" [75].

In their attempt to implement a causal inference computational framework (now available as an open-source python package [288]), Microsoft writes plainly about the challenges that motivated the development of their code library [289]:

- Empirical causal inference is "daunting" despite the knowledge available in the literature [289]
- Ensuring that underlying assumptions are properly identified and validated is also "daunting" [289]
- Causal inference depends on the estimation of unobserved quantities, which itself depends on assumptions about the data-generation process [289]

Although their code streamlines this process a bit, it does not inherently solve the fundamental problems in causal inference. As with all mathematics, the validity and breadth of one's underlying assumptions determines one's ability to properly identify causation and model the effects that follow. Unfortunately, the use of assumptions is inescapable (otherwise, it is impossible to scope an investigation). These remarks by Microsoft, a well-resourced institution of talented scientists and engineers, temper our expectations regarding causal inferencing in general, as well as the ability to generalize any bottom-up modeling technique that explains causation across multiple levels of abstraction. Microsoft's observations are consistent with arguments by Schmidt [156] and Richardson [155] that despite any framework's ability to represent interacting components, only careful and thoughtful iterative experimentation will ultimately determine a model's validity,[375] and it will not always be possible to validate a model in a straightforward manner, even when the model is valid.[376]

Within causal inference studies, selecting the right variables is paramount. The same is true for model discovery.[377] This thesis, however, is not geared towards model discovery. Therefore, the questions in this thesis are not, "What is the true model?" but rather, "Do models of high-level behavior built with high-level variables explain more of that behavior than models built with low-level variables?" In this thesis, the simulation serves as the true model, and so the burden of finding an "equally true" emergent behavior model is lighter than it might otherwise be in other applications. CHAPTER 7 and CHAPTER 8 will study the extent to which it is possible to get away with emergent behavior exploitation in the absence of the true emergent behavior model by targeting

---

[375] Readers interested in the subject of empirical validation are referred to a great article by Dougherty [268].
[376] Models of chaos theory are typically valid in this sense. It is difficult to faithfully reproduce a simple, empirically observed chaotic trajectory in a computer, and impossible to do so for all trajectories.
[377] To an engineer, there is little difference between model discovery and causal inference, but that is partly because the causal linkages are usually the responsibility of the engineering professional, rather than being built into the mathematical notation of the model and structure of the model formulation process.

component properties as well as properties associated with the self-organized collection. The essence of this argument is that it is easier to identify an emergent behavior than it is to accurately model one (to derive the governing function describing the interaction, whatever its form may be). One can determine that a property is emergent if an adequate set of approximate models is found. However, if model discovery were the goal, or even if emergent behavior design were the goal, an experimentation step would be required to validate the model on a real system,[378] and a broader set of model generation tools would be needed to explore the space of candidate functions.

Thanks to efforts by researchers around the world to provide open-source tools over the past 10-15 years, it is now relatively easy to create a very powerful, executable SE modeling environment for complex behavior, and SoS modeling. Physical and functional decompositions can be created using Python code that is open source and UML/SysML compliant [290]. The limitations of standard decompositions can be overcome with any one of several hypergraph libraries (HALP is in Python [291], and others can be found at [292] [293] [294] [295]). Those hypergraphs can then be interfaced with modeling software as well as causal inference methods [288] using Graph Neural Nets [296] (also an open source Python package [297]). A variety of simulation-based inference methods are reviewed in [298]. The System Dynamics Society provides links to open-source system dynamics models in R and Python [299]. A Discrete Event Simulator is also available in Python [300]. For simulations of objects that obey known physical laws, see also [301]. Graph nets can be used in a workflow to produce symbolic regressions as in work by

[378] Again, “validation” here may be limited to a more qualitative sense. For example, the model predicts chaotic behavior, and the real system exhibits chaotic behavior that conforms to that same model (although it is impossible to validate the coefficients of that model with infinite accuracy).

Cranmer [302] (see Appendix for discussion of symbolic regression). The methods in this thesis are readily incorporated into such a workflow to mine the data for emergent behavior (using even more sophisticated pattern recognition algorithms, see Section 5.2 and [303] [304] [305]) as part of a larger system modeling and simulation environment. The pattern recognition and behavior association steps of this thesis would be performed on the output of simulation data, which would then alert decision makers to the presence of self-organization and candidate emergent behaviors. Furthermore, a feedback mechanism can be built into the environment to update the SysML diagrams whenever self-organization and/or emergence are identified (this would require merging the SysML and hypergraph capabilities). The data throughput of such an environment can be made more efficient using any number of surrogate modeling, data compression, and machine learning techniques, and design considerations can be incorporated by mapping requirements to a constrained optimization algorithm module. The exploitation analysis step would be interfaced with the optimization scheme to provide recommendations for ways to modify system design (the approach for exploitation analysis will be illustrated in CHAPTER 7-CHAPTER 8). While this opportunity is exciting, it represents an additional 12-24 month code development process, and so, it is future work. Readers considering taking on this or a similar approach are referred to guidance on MBSE by Hause [306].

A.11 Brief Review of Other Symbolic Regression Tools

The standard approach in regressions since at least the 1970's was to assume the form of the function and fit the coefficients of the function to the available data until a suitable model is found (for example, entire methods have been built on cases where a polynomial can adequately fit a region of data [307]). Shortly thereafter, universal approximators such as artificial neural networks gained wide use [308], with the philosophy being that if the true form of the function need not be known, then it is better to have a flexible function that can approximate any data set. Note, however, that an ANN achieves its flexibility through repeated application of linear combinations and compositions of the same function. Now that computers are powerful enough, researchers have shifted to studying the alternative to these approaches: rather than assuming a particular functional form and fitting the coefficients, consider instead a variety of functional forms (ranging from closed-form expressions to partial differential equations[379]). This approach has been called ***symbolic regression***. PySINDy appears to be intended well suited for long time-series and produces partial differential equations as outputs [309] (the time-series in this thesis vary in length, but are generally much shorter than the examples given for PySINDy). Work by Cranmer [302] incorporates deep learning to guide the symbolic regression process. Both AI Feynman v1.0 [310], and 2.0 [311] produce a wide range of symbolic regressions, while AI Poincaré [312] generates symbolic regressions for conservation laws (as does work by Schmidt & Lipson [313]). Most notably, AI Feynman considers the time-complexity of the expression in its model-selection scheme as well. However, their computation of bit complexity assumes finite precision representations, and counts parametrized terms within equations along with the operators themselves (e.g. in their kinetic energy example, the mass and velocity variables are counted along with the multiplication and division operations).[380]

[379] It does not yet seem that any work has been done towards fitting integral equations to a set of data.

[380] A study comparing the various information theoretic and algorithmic complexity approaches is warranted, but outside the scope of this work.

A.12 Contrasting with Other Behavior Association Studies (Extended)

Dr. Balestrini-Robinson's PhD thesis [81] provides an interesting review of Complexity Science as well as a thorough review of the state of M&S. While his work references self-organization and emergence, he does not tackle the topics individually. Rather, his discussion centers largely on the concept of "organized complexity" and how this distinguishes complex systems from disorganized systems amenable to statistical mechanics (massive numbers of atoms) or "simple systems" with few systems and few variables (the discussion leans on work by Weaver; see [81]). His thesis then turns towards determining whether useful information about the behavior of large, complex systems can be obtained without a complete simulation of the system's components and their interactions, and, if such a simulation is needed, determining which components are the most important to simulate and which behaviors are the most important to model in detail. To do this, Balestrini-Robinson relies on knowing the functional decomposition *a priori* (a graph that he shows to be layered, directed, and cyclical). Furthermore, he envisions an approach that is iterative where his Digraph modeling technique (DiMA) can be used to identify a lack of information in the simulation environment, which decision-makers can then use to modify said environment. His approach could be used in conjunction with the environment by Cummings, and makes the contribution of helping decision-makers avoid running unnecessary simulations. However, DiMA does not provide guidance on how many new behaviors have been omitted, or how to identify/derive them.

Work by John Collier [314] provides a list of sufficient (and necessary) conditions for emergence as well as a strong rebuttal to Kim's causal exclusion argument. He characterizes emergent properties as "not … reducible to the binary relations among [the] components, [and] unpredictable from the properties of their compositional substrate, and to show new or novel properties that do not exist in their substrate" [314]. Although these appear to be similar to the concepts in this thesis, Collier's condition of unpredictability seems a bit too strong, since it requires that the emergent property not be numerically computable. Collier associates emergence with dissipative systems, which is a common theme in the literature on emergence (see review by Kitto [73]). This quality has been omitted from this thesis because it is largely taken for granted in engineering. Furthermore, it is unclear how a non-dissipative (and non-physical) construct like a perpetual motion machine would impede self-organization and emergence.[250] He then goes on to list five sufficient conditions for emergence:

> *1. The system must be nonholonomic, implying the system is nonintegrable (this ensures nonreducibility)*
> *2. The system is energetically (and/or informationally) open (boundary conditions are dynamic)*
> *3. The system has multiple attractors*
> *4. The characteristic rate of at least one property of the system is of the same order as the rate of the non-holonomic constraint (radically non-Hamiltonian)*
> *5. If at least one of the properties is an essential property of the system, the system is essentially non-reducible; it is thus an emergent system* [314]

The condition of non-integrability, once again, seems too strong.[381] That the system is open is trivial in all real engineering applications except maybe quantum computing. Collier considers condition 3 to be debatable, and so it will not be discussed further. The fourth condition may serve as a useful condition, except that it assumes at least one property of the system has the same units of measure as the non-holonomic constraint. This imposes a

[381] It may be a simple confusion due to semantics, but a partial differential equation that cannot be "integrated" is an equation that has no solution. To an engineer, an equation that has no solution is equivalent to a non-physical equation (an equation that is wrong / incompatible with reality).

strong criterion on the system of equations describing the system. Future mathematical derivations will prove whether this overly constrains the system of equations. Regarding the fifth condition, Collier does not elaborate on what it means for a property to be "an essential property of the system," but it appears to be analogous to the direct interaction criterion discussed in Section 4.3.4. Furthermore, Collier's example of an emergent property is the fact that Mercury is caught in a 3:2 spin-orbit resonance due to the Sun. Collier does not explain how this numerically or phenomenologically impacts any other part of the solar system (or object within the solar system), and so his exposition does not have a clear tie to functional emergence. Collier's example is much closer to self-organization than emergence, by the definitions in this thesis.

An interesting paper by Moyal, Fekete, and Edelman [315] on the study of human cognition as an emergent behavior describes their Dynamical Emergence Theory. While the data and metrics discussed by the researchers are too far beyond the expertise of this author to scrutinize in detail,[382] the concepts and tools they employ are consistent with the overall approach in this thesis. The researchers write, "we seek… a functional and computational understanding of the relationship between the structure of a system's collective dynamics … and that of the [emergent behavior] it is capable of producing" [315]. A particularly noteworthy argument is the reiteration that their definitions[383] (as those in this thesis) "suggest a natural … partitioning of any nontrivially structured dynamical system into functional levels of organization, based on the extent to which information about some components' time series is encoded in others" [315]. They then go

[382] A more thorough review of this paper is also handicapped by time constraints.
[383] It has already been demonstrated throughout this thesis how many articles use the same terminology to describe different phenomena. Time does not permit a thorough disambiguation.

on to reference algorithms one can use to quantify causal influence in nonlinear systems as well as time-series motif identification[384] (the interested reader is encouraged to review those, as they are mostly distinct from the methods listed thus far in this thesis). While their work outlines necessary conditions for emergence, only one is of interest here (the other two are in direct relation to consciousness itself): "Effectiveness: … [an emergent behavior] is a functional (as opposed to epiphenomenal) trait … its states and transitions are causally and predictively effective" [315]. This is essentially a refutation of Kim's causal exclusion argument. To an engineer, the equivalent would be: there is no problem with saying that an unbalanced force causes a mass to accelerate, whether that mass is a quark or a car, and whether the force is the strong/weak force or the friction between the road and the tires. In each case, the assumption is the same: the idealized basic component is an indivisible body with simple properties that are predictable and persistent (see Section 2.2.2). In each case, the assumption will fail if pushed beyond its intended use. Both levels of abstraction contain equally valid descriptions of *causation*. The researchers strongly indicate that their method assumes self-organization precedes emergence (as understood in this thesis), and then proceed to justify their behavior association step by arguing for the predictive effectiveness of their quantitative metrics.[385] Without delving into the details, this is an example of subject-matter experts positing the properties of their higher-level object and then creating the link between those properties and the dynamics of the self-organized object (the physical brain). Within the context of this one paper, their approach

[384] The aforementioned Matrix Profile is one such algorithm. "Motifs" are patterns that appear frequently in time series, which have clear application in the identification of self-organized structures. See also [392].
[385] For clarity, the metrics are: representational capacity, amount of experience, and nature of experience. They're stated here so the critical, well-informed reader can determine if a mistake was made in the arguments provided above.

does not outline a systematic method for emergent-behavior discovery,[386] but proceeds to argue that the associations they find between brain electrochemical dynamics and their three high-level metrics characterize the functional essence of human awareness. In order to do this, they perform experiments wherein they stimulate the brain and associate measurable changes in the brain's electrical activation with states of awareness, as represented by their quantitative metrics. That is, they search for evidence of interactions, as in this thesis, but in their case, the external stimulus is affecting the quantitative properties of awareness rather than attempting to model how the quantitative properties of awareness correspond to actions undertaken by the human being possessing it.

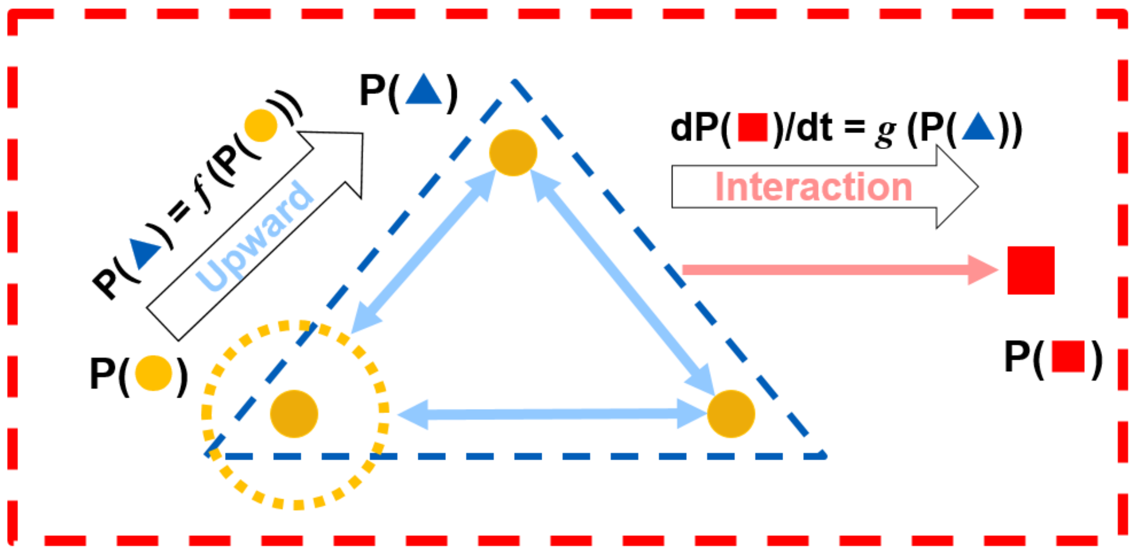


Figure 125 – Behavior association conceptual diagram (simplified Figure 21, step 2)

If their metrics do not completely map to awareness, or if they map to other traits in addition to awareness, then their theory runs the risk of producing an association fallacy. That risk is inherent in every study of emergent behavior. More importantly, by positing the

[386] They authors may have done so elsewhere.

quantifiable properties of the emergent behavior in question the researchers attempt to manually overcome[387] the modeling challenges of emergent behavior discovery,[388] which will be discussed for the remainder of this subsection.

Real "components" in isolation have many behaviors that are unobservable, and therefore unknowable,[389] outside of a specific context. For example, a bee studied in near-total isolation (e.g. the vacuum of space) would die quickly, and nothing about its behavior would be learned from observation, except a process by which it dies. Now suppose we have idealized[390] components with finitely-many, known properties (depicted in Figure 125 as yellow circles), and a similarly-idealized system external to those components (represented by a red square). The idealized objects can be anything (atoms and a molecule, birds and a flock, etc.). The structure of the red square is neglected for convenience. If the circles are studied in isolation (dotted yellow circle), the only information that can be obtained is that they each have a set of properties P(●). No emergent behavior can be detected within such a limited scope. Similarly, the red square has properties P(■). Now suppose a simulation is constructed such that three circles can interact. Due to their two-way interaction rules[391] and under the right circumstances, they self-organize into a system indicated by a dotted blue triangle. This system has a set of properties, P(▲), which, in the worst case, contains infinite properties. Since nothing else is known about the system, all

387 All subject matter experts do. This author is no exception.
388 To this author's knowledge, no other work has named the problem thus.
389 Within the confines of modern science.
390 See Section 2.2.2. Note, however, that the empirical equivalent of an idealized component is the set of measurable properties taken in a physical experiment (based on the availability and precision of measurement devices, choice of control variables, and other experimental limitations).
391 The two-way interaction serves no purpose in this example other than to illustrate the kinds of interactions that exist. Examples of two-way interactions include interatomic forces, and social bond.

of its properties are arbitrary.[392] Within this scope (the dotted triangle) it is impossible to observe any emergent behaviors. Furthermore, there are at least multiple, if not infinitely-many, possible functions, *f*, that map those system-level properties to component level properties, P(●). So, the first of the modeling challenges with emergent behavior discovery is that before any emergence has occurred, the self-organized object already possesses infinitely many arbitrary properties.

The remaining challenges stem from issues common to all regressions, but take on special significance in this context. Recall that the presence of interactions is the strongest evidence a simulation can provide for the existence of an emergent behavior. Now, consider a simulation with a large enough scope to detect emergence (the red dotted box), where the triangle is exposed to the red square and can interact with it. This means that functions, *g*, can now be written that map changes in the red square's properties over time to the properties of the triangle (and there may be infinitely many such functions). In principle, this turns the emergent behavior discovery problem into an inverse design problem, where any property in P(▲) can be found by taking the inverse function of the interaction equation involving some property in P(■).[393] However, it is unlikely that the functions will be invertible.[394] Then, for any finite data set (i.e. all practical data sets) there

[392] Depending on the properties of the components, any number of high-level properties can be constructed: geometric, graph-theoretic, spectral, biological, mechanical, chemical, etc.

[393] Most upward causation models are not invertible because they are multivariate.

[394] In fact, this provides another simple explanation for why so many researchers have associated emergent behavior with the element of surprise / unpredictability: since the mapping between form and function is not 1:1, the emergent behavior models are not invertible. Since they are not invertible, it is very difficult, if not impossible, to predict what the emergent behavior will be from the knowledge of the behavior of the components. However, surprise, in and of itself, is not a genuine quality of emergence. Once the emergent behavior is observed, it can certainly be reproduced / predicted, for the same reason that a function can be rendered piece-wise invertible (so to speak) by carving the domain into compact intervals where the function is 1:1 over the interval. Furthermore, a very clever person can certainly envision some emergent behaviors. Every inventor in history certainly has.

will always be a set of deceptive functions that relate the red square properties to the blue triangle properties with arbitrarily small, non-zero error that will be very difficult to distinguish from models that are essentially causal but contain some small fitting error.[395] Finally, if there exists some exact model[396] relating the red square properties to the properties of multiple yellow circles, this raises the question of whether one must accept that function as a property of the blue triangle, regardless of whether it conforms to some known property or not. In other words, the question is: should all exact models of red square interactions with multiple simultaneous yellow circles be attributed to the blue triangle? A simulation, alone, can only answer to the affirmative because it has no alternative hypothesis to test against.[397] This will probably contradict human experience and intuition more often than not. The question can only be settled empirically. In short, the main modeling challenges with emergent behavior detection are:

1. A self-organized object can have infinitely many arbitrary properties
2. There are infinitely many functions that map an arbitrary property to the time-rate-of-change of a known system property (i.e. interaction equations can be written to arbitrary accuracy, whether or not they are truly causal)
3. A single simulation (regardless of how many times it is executed) attributes all multi-component interactions to the system made up of those components

The first two challenges demand more than a goodness of fit to make the argument that the property of a self-organized object is truly emergent (not arbitrary). CHAPTER 4 of this

[395] Here, a deceptive model is equivalent to a nonlinear model containing the wrong variables yet have low error, as opposed to a model that contains the right variables but is not quite correct in its form or coefficients. This presupposes that true causal information exists within the data set. See Assumption 1 in Section 2.2.
[396] "Exact" meaning that its predictions hold for all extrapolations.
[397] Properties defined this way fall under the strict distinctiveness criteria in Section 4.3.3.

thesis provides conditions to help filter down this list of possible candidates, whereas in the work by Moyal, Fekete, and Edelman [315], the researchers implicitly address these challenges by positing a set of three quantitative properties attributable to awareness (based on their knowledge and experience) and then test it empirically. Note that although the third challenge is attributed to simulations, it can be extended to experiments as well (multiple experiments for each property are needed to test against false positives / negatives).

A.13 Emergence and Self-Organization as Latent Nodes

Rather than rely on long philosophical arguments, engineers may find it intuitive to simply extend the concept of latent variables to physical and functional decompositions, and from those graphs identify where the useful information is obtained.[398] First, consider a few analogies. In mathematics, the classes of problems that are best understood are linear problems (be they linear systems of algebraic equations, linear partial differential equations, etc.), but the majority of problems are nonlinear. Similarly, in engineering, the best understood fluid mechanics problem are laminar flows, but most real flows are turbulent. In SE, the best understood systems are decomposable in their functions. So if the expectation is that most systems are complex (the majority of natural systems certainly are), what complicating feature do their graphical decompositions possess that standard systems do not have? Building on the concept of latent variables, perhaps the expectation should be that every decomposition will likely contain one or more latent nodes (i.e. every

[398] The reader is referred to [398] for a more sophisticated use of latent nodes. This thesis can proceed with a human-in-the-loop approach due to the nature of the hypotheses presented. To render these ideas compatible with SE current practices, a couple of starting points are [399] [286] [57] and their references. To place these steps within the broader CBA see [56] and her references, as well as Slide 5 in [400].

decomposition is assumed to be incomplete until proven otherwise). For physical decompositions, these nodes represent latent objects (the structures caused by self-organization), while in functional decompositions, these represent latent functions (emergent behaviors). In this sense, a primary task in every system design project is to root out the latent nodes. An example of this concept is given in Figure 126.

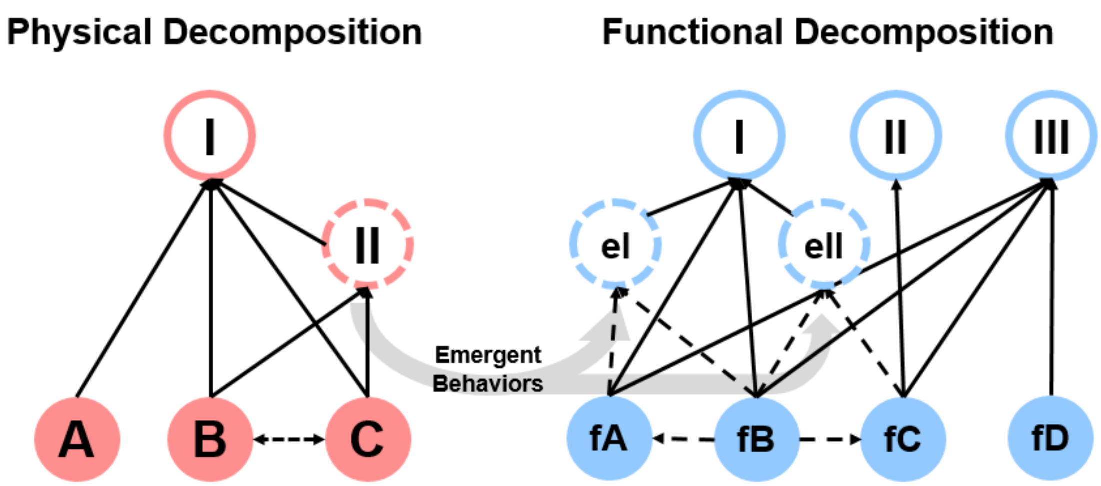


Figure 126 – Notional physical and functional decompositions as directed, layered graphs with latent nodes

Figure 126 depicts a physical decomposition that contains a self-organized system, and a functional decomposition that contains two emergent behaviors.[399] Suppose System I is the system the engineer intended to design using components A, B, and C, while System II is the system that arises when components B and C self-organize. Here, System II is treated as a component of System I because the complete set of components is always attributed to the intended system by convention.[400] System II possesses emergent behaviors, depicted

[399] A textbook functional decomposition would include an "and" node (called an "and gate" [408]) for coupled functions. The interested reader is referred to Melançon's very informative Master's thesis [407].
[400] It is not difficult to imagine a notation where the intended system as well as the unintended self-organized systems are all placed at the same level in the graph (provided they're made of the same components).

on the right in Figure 126 (clearly these behaviors are also attributable to System I, but this need not be the only convention). Those emergent behaviors (e I, II) are also latent nodes, and those emergent behaviors make a direct contribution to one of System I's primary functions (Function I). The faded gray arrows indicating the association of emergent behaviors with the self-organized system are provided for exposition only. Component-level functions are denoted f A, f B, f C, and f D. Lower level function fB is depicted as influencing functions fA and fC. The latent *nodes* and the edges associated with them are dotted. The dashed nodes are latent *systems* in the physical decomposition, and latent *behaviors* in the functional decompositions. Some edges are dashed to help distinguish their meaning from the standard edges in each graph. In the physical decomposition, a dashed edge means interaction while a solid edge means membership. In the functional decomposition, a dashed edge means coupled cause, while a solid edge means independent cause. It is supposed that the emergent behaviors are independent causes for the sake of this example.

The concept of the latent node is an extension of the following argument. Systems that are physically decomposable are those where the arrangement of the components is either irrelevant or unique. Therefore, a system that is not physically decomposable is one where different arrangements result of components result in markedly different systems. In this way, self-organization results in a latent node in the physical decomposition. Analogously, systems that are functionally decomposable are those where the arrangement of functions (in time) is irrelevant or unique. This corresponds to cases where the functions are executed independently and sequentially. However, if the functions are "arranged" into different processes (they become coupled, executed concurrently, etc.), that new process may result in a new higher-level function that would not otherwise exist. Such a function would be an emergent behavior, that it would be a latent node in a functional decomposition.

A.14 Additional Comments Relating to Tests of Hypothesis 2

Given the concept of modeling the emergent behavior put forward in this chapter, the concept of weak emergence raises one important question: if the only way to accurately observe emergence is by running the full simulation, how can one expect to accurately model it using some function other than the simulation itself? Perhaps this contradiction is purely a superficial one. For example, macroscopic temperature is only meaningful when the system components are in a quasi-equilibrium or true equilibrium state. If they are not, it is physically and mathematically impossible to accurately estimate a useful mean value (recall that temperature is a local mean value). So we see that, if the underlying conditions are correct, history shows it is possible to make a useful model that predicts a certain class of system-level behavior and the interactions between that system and other systems that contain related properties (pressure, density, etc.). What the equations of thermodynamics cannot do is predict their own failure (this is a job typically relegated to the engineer, medical professional, etc.). There is no information within the classical thermodynamics equations that predicts when the atoms in the substance are no longer in quasi-equilibrium, because the continuum assumption has discarded that information. Therefore, if the question we are asking is "What causes emergent behavior X?" or "When/how can we observe the manifestation of emergent behavior X?" then we absolutely must use the bottom-up simulation, because only the simulation can predict all possible behaviors of the components and predict when those components will settle into a stable-enough form to yield useful system-level quantifiable properties.[401] Simply put, macroscopic laws cannot say anything about emergence (with respect to their own components) because they assume the emergence is always present. In this sense, macroscopic laws are a kind of curve-fit that bring with them the information loss engineers have come to expect from all regressions.[402]

A number of ways to test Hypothesis 2 were considered. The tests in Section 5.5.2 are special cases of the ways considered below. Due to their simplicity and effectiveness, the tests were restricted to those special cases. For the sake of completeness, the full discussion is provided below. One can falsify Hypothesis 2 by:

1. Attempting to fool the numerical criteria using nonlinear transformations of the properties of a single component within a system
2. Attempting to fool the numerical criteria by treating the properties of a single component contained within a system as interchangeable with those of the system
3. Critical examination of the predictions of the numerical criteria under "normal" operations

Recall that weak emergence merely requires that the simulation be run in order to recognize and quantify the property in question.[403] All properties associated with self-organized

[401] Furthermore, if this were not the case, artificial neural networks could not be called "universal approximators."
[402] In defense of continuing the use of emergent behavior terminology, perhaps "information loss" is too strong a term. It would be more accurate to say "information trade-off" where the macroscopic dynamics becomes readily understood at the expense of lower-level details, and vice-versa.
[403] Referring to the example by Abbott [72], the simulation does not need to be run to determine that the interior angles of a triangle sum to 180 degrees. However, it does need to be run to observe that the components sometimes arrange themselves into a triangle, and then to determine that one or more of those angles are significant in some way.

entities satisfy that criterion. This trivial consequence of self-organization inspired the "relaxed distinctiveness" definition in Section 4.3.3. The relaxed distinctiveness definition makes it impossible to define a single-component system in the naïve sense. However, the relaxed distinctiveness definition does permit a single-component system if the system-level properties are arbitrary nonlinear functions of the properties of a single component. This is an absurd situation and defeats the purpose of making a distinction between low levels and high levels.[404] Furthermore, the simulation would no longer be needed because the properties can be computed from the values of the component properties, which violates the definition of weak emergence. Take, as a simple example, the x-coordinate of a boid in a simulation $P(A) = x$. If something as trivial as the square of that value satisfies the numerical criteria, as in $P(SoS) = x^2$, then there is a contradiction and the hypothesis is false. The potential for this exists in both the strict and relaxed definitions. Therefore, this test would see if the criteria can be fooled using nonlinear operations on the properties of a single component. The intention of the distinctiveness criteria was to ensure at least one system-level property was a mathematical function of multiple component properties at once. In this way, no single component constitutes the basis for every system-level property, and so it becomes meaningful to distinguish between the two levels (the expectation being that, in practice, most quantifiable system properties will require multiple component inputs). However, this test cannot proceed effectively without firm knowledge of the upper bound on the number of emergent behaviors a system can have (i.e. Hypothesis 1 would have to be proven true beyond doubt). Once Hypothesis 1 (or

[404] The intention was that the number of systems in the definition be $n > 2$, but since it was not clearly stated, the opportunity for confusion exists, hence the experiment. Furthermore, this presents the opportunity to study whether merely imposing $n > 2$ resolves the fallacy, or whether a deeper issue exists.

some equivalent formula) is proven valid, then it will be possible to search for meaningful distinctions between a handful of arbitrary nonlinear properties and an equal number of apparently meaningful emergent properties, in order to create stronger criteria.

The approach would attempt to fool the numerical criteria by probing the limitations of the definition of association. There are some cases (typically linear properties) where an emergent property can become quantitatively equal to the property of a component (for example, if the heading of a flock is defined as the mean of the headings of its birds, then when their headings are constant, the values are equal). To a numerical algorithm performing behavior association, the two time series would be largely indistinguishable. Therefore, the question becomes, "can the numerical criteria be fooled by cases where the property assigned to a stable system is numerically equivalent to the property assigned to its components?" This question is nullified by the distinctiveness criteria in CHAPTER 4. Since swapping the variables would not work at every iteration in the time series for all simulation, it suffices that the equations used to compute properties be different, even if the name of the property is the same or the values occasionally match.

All of these tests rely on the implicit assumption that if an interaction occurs between a higher level object and anything else in the simulation[405] then the properties in that equation are performing a function. However, it is unusual for engineers to attribute a function such as collision avoidance to a flock (as opposed to the birds themselves). Logically, the two can be conflated in this thesis because of Assumption 1 (every behavior

[405] The focus here is on objects comprised of dynamic agents. The math in this thesis would have to be altered slightly to accommodate obstructions that are typically classified as part of the environment (such as buildings).

of the flock is *bona fide*). Practically, and more importantly, the two can be conflated because if an interaction is predictable it is also exploitable and, in this way, creates a latent function that the engineers may not have intended (see Appendix for more information on latent node representations). The ability to make this claim in this thesis is due to Assumption 2 (there are no missing variables that would make exploitable behaviors in a simulation unattainable in reality). Therefore, when claiming that an interaction has been identified, this thesis will operate as though that suffices to claim that a higher level function has been identified. If this tacit assumption becomes problematic, it will be discussed in the appropriate results section. Again, to clarify: this generosity towards the validity of the model is permissible only because the numerical criteria cannot work in a real-world case if they do not first work in a simulation.

The third approach is broader than a simple check for a logical fallacy or deceitful statistic. While keeping in mind that Hypothesis 2 is a sufficient condition, the third approach will place the emphasis on whether or not the numerical criteria seem trustworthy (for lack of a better term). Clearly, "seeming trustworthy" is deeply subjective. Nevertheless, since the properties considered in this thesis will mostly be geometric properties of the self-organized shape (many of which are linear combinations of properties), asserting that those properties are emergent is not outlandish. Therefore, the problem of trustworthiness[406] lies in the whether or not the conclusions drawn from the data are robust. For that, this thesis can extend a non-controversial way of testing a data set for trustworthiness based on best-practices within engineering and computer science [247]. Specifically, the interaction models obtained on one data set can be extrapolated to other

[406] That is, one of the problems considered here.

data sets to see if the variation in error changes the outcome of the analysis. Since this feature seems essential to any regression technique, checking for bad extrapolation was built in to the behavior association step.

REFERENCES

[1] Defense Acquisition University, "DAU Glossary," 15 July 2019. [Online]. Available: https://www.dau.mil/glossary/SiteAssets/download.aspx. [Accessed 15 July 2019].

[2] M. J. Sullivan, T. Masters, P. Anderson, M. Bonner, R. Bullock, K. Gurka, S. Hiromoto, J. McSween, R. Miller, L. Parkey and T. R. Sun, "Weapon System Requirements: Detailed Systems Engineering Prior to Product Development Positions Programs for Success," United States Government Accountability Office, Washington D.C., 2016.

[3] Office of the Secretary of Defense, "Quadrennial Defense Review (QDR) Report," Department of Defense, Washington D.C., 2010.

[4] C. M. Strube and J. R. Loren, "Portfolio Influences on Air Force Capabilities-Based Assessment and Capabilities-Based Planning Activities," in *6th International Conference on System of Systems Engineering*, Albuquerque, 2011.

[5] A. F. Lykke, "Defining Military Strategy: Strategy = Ends + Ways + Means," *Military Review*, vol. 69, no. 5, pp. 2-8, 1989.

[6] J. A. Drezner, M. V. Arena, M. McKernan, R. Murphy and J. Riposo, "Are Ships Different? Policies and Procedures for the Acquisition of Ship Programs," RAND Corporation, Santa Monica, 2011.

[7] Defense Acquisition University, "Chapter 1 - Program Management (Defense Acquisition Guidebook)," 26 February 2017. [Online]. Available: https://www.dau.mil/tools/dag. [Accessed 17 July 2019].

[8] J-8, Joint Staff, "Manual for the Operation of the Joint Capabilities Integration and Development System," Department of Defense, Washington D.C., 2018.

[9] Joint Staff, Charter of the Joint Requirements Oversight Council (JROC) and the Implementation of the Joint Capabilities Integration and Development System (JCIDS), vol. CJCSI 5123.01H, Washington D.C.: United States Department of Defense, 2018.

[10] L. G. Shattuck, "The Joint Capabilities Integration and Development System and the Capabilities-Based Assessment," [Online]. Available: http://faculty.nps.edu/dl/hsi_certificate_program/oa3412/module03/print_versions/mod03_jointCapabilities2.pdf. [Accessed 24 September 2018].

[11] J. F. Schank, "Analysis of Alternatives: Keys to Success," in *Proceedings of the Ninth Annual Acquisition Research Symposium - Volume I*, Seaside, 2012.

[12] P. K. Morrow, "Teaching Note: Analysis of Alternatives," Defense Acquisition University, Fort Belvoir, 2011.

[13] M. Schwartz, "Defense Acquisitions: How DOD Acquires Weapon Systems and Recent Efforts to Reform the Process," U.S. Congressional Research Service, RL34026, 2014.

[14] R. S. Johnson, "The Changing Nature of the US Navy Ship Design Process," *Naval Engineers Journal,* vol. 92, no. 2, pp. 88-113, 1980.

[15] C. Graham, M. Bosworth and D. Taylor, "Designing the Future U.S. Naval Surface Fleet for Effectiveness and Producibility," in *The National Shipbuilding Research Program 1989 Ship Production Symposium*, Arlington, 1989.

[16] J. F. Schank, S. Savitz, K. Munson, B. Perkinson, J. McGee and J. M. Sollinger, "Designing Adaptable Ships: Modularity and Flexibility in Future Ship Designs," RAND Corporation, Santa Monica, 2016.

[17] J. Mun, T. Housel and L. B. Majchrzak, "Business Case Valuation of Strategic Flexibility in Shipbuilding: Justifying and Assessing the Value of Flexible Ships Design Features in New Navy Ship Concepts," Naval Postgraduate School, Monterey, 2017.

[18] C. A. Largiader, "Modular Platform Based Surface Ship Design," MITLibraries, Cambridge, 2001.

[19] M. J. Corl, "Methodology for Optimizing Commonality Decisions in Multiple Classes of Ships," University of Michigan, Ann Arbor, 2007.

[20] M. Smidt and M. Junge, "A Modular Warship for 2025," *Proceedings Magazine*, vol. 140, no. 1, January 2014.

[21] R. G. Keane and H. Fireman, "Producibility in the Naval Ship Design Process: A Progress Report," in *Proceedings of the Ship Production Symposium*, New Orleans, 1992.

[22] D. R. Mensh, R. S. Kite and P. H. Darby, "The Quantification of Interoperability," *Naval Engineers Journal*, vol. 101, no. 3, pp. 251-259, 1989.

[23] S. A. Brown, Ed., Providing the Means of War: Historical Perspectives on Defense Acquisition 1945-2000, Washington D.C.: United States Army, 2005.

[24] Office of the Chairman of the Joint Chiefs of Staff, DOD Dictionary of Military and Associated Terms, Washington D.C.: Department of Defense, 2019.

[25] D. Carney, D. Fisher, E. Morris and P. Place, "Some Current Approaches to Interoperability," Carnegie Mellon University, Pittsburgh, 2005.

[26] T. C. Ford, "Interoperability Measurement," Air Force Institute of Technology, Wright-Patterson Air Force Base, 2008.

[27] E. A. J. Wyatt, "A Reliability-Based Measurement of Interoperability for Conceptual Systems of Systems," Georgia Institute of Technology, Atlanta, 2014.

[28] M. Collette, "Hull Structures as a System: Supporting Lifecycle Analysis," *Naval Engineers Journal*, vol. 123, no. 3, pp. 45-55, 2011.

[29] S. Bernstein, "Methodology for Interoperability-Enabled Adaptable Strategic Fleet Mix Planning," Georgia Institute of Technology, Atlanta, 2017.

[30] T. A. Laverghetta, "Dynamics of Naval Ship design : a systems approach," Naval Post Graduate School, Monterey, 1998.

[31] D. N. Ford and A. Clark, "Modeling the Department of Navy Acquisition Workforce With System Dynamics," in *Proceedings of the Fifteenth Annual Acquisition Research Symposium - Volume I*, Monterey, 2018.

[32] R. W. Button, "Thinking Constructively About Overmatch," RAND Corporation, 2 August 2019. [Online]. Available: https://www.rand.org/blog/2017/03/thinking-constructively-about-overmatch.html. [Accessed 21 March 2017].

[33] C. Pellerin, "Work: DOD embracing innovation to fuel military overmatch against adversaries," DoD News, 4 May 2017. [Online]. Available: https://www.army.mil/article/187213/work_dod_embracing_innovation_to_fuel_military_overmatch_against_adversaries. [Accessed 2 August 2019].

[34] M. T. Klare, "Why 'Overmatch' Is Overkill," The Nation, 20 December 2018. [Online]. Available: https://www.thenation.com/article/overmatch-pentagon-military-budget-strategy/. [Accessed 2 August 2019].

[35] C. Pellerin, "Deputy Secretary: Third Offset Strategy Bolsters America's Military Deterrence," DoD News, 31 October 2016. [Online]. Available: https://dod.defense.gov/News/Article/Article/991434/deputy-secretary-third-offset-strategy-bolsters-americas-military-deterrence/. [Accessed 2 August 2019].

[36] Office of Aerospace Studies, Capabilities-Based Assessment (CBA) Handbook: A Practical Guide to the Capabilities-Based Assessment, Kirtland AFB: Air Force Materiel Command (AFMC) OAS/A5, 2014.

[37] N. Doerry and H. Fireman, "Fleet Capabilities Based Assessment (CBA)," *Naval Engineers Journal,* vol. 121, no. 4, pp. 107-116, 2010.

[38] P. Koenig, D. Nalchajian and J. Hootman, "Ship Service Life and Naval Force Structure," in *ASNE Symposium Engineering the Total Ship,* Tysons Corner, 2008.

[39] D. A. Rains, "Fleet Mix Mission Effectiveness Analysis," *Naval Engineers Journal,* vol. 111, no. 1, pp. 65-81, 1999.

[40] T. Somboonwiwat, "A Methodology to Solve Large-Scale Group Fleet Replacement Problems," Oregon State University, Corvallis, 2001.

[41] P. C. Koenig, P. M. Czapiewski and J. C. Hootman, "Synthesis and analysis of future naval fleets," *Ships and Offshore Structures,* vol. 3, no. 2, pp. 81-89, 2008.

[42] C. Hawkinson, "Appendix Fourteen: United States Navy Fighter Tactics," Internet Archive Wayback Machine, 4 February 2012. [Online]. Available: https://web.archive.org/web/20120204195154/http://www.centuryinter.net/midway/appendix/appendixfourteen_usvftac.html. [Accessed 13 August 2019].

[43] D. Parsons, "Naval Aviation in WWII: Fighter Tactics in WWII," *Naval Aviation News,* vol. 75, no. 5, pp. 30-35, July-August 1993.

[44] L. Dwyer, "Grumman F4F Wildcat," The Aviation History Online Museum, 19 February 2014. [Online]. Available: aviation-history.com/grunman/f4f.html. [Accessed 14 August 2019].

[45] L. Dwyer, "Mitsubishi A6M Reisen (Zero-Sen)," The Aviation History Online Museum, 25 March 2014. [Online]. Available: aviation-history.com/mitsubushi/zero.html. [Accessed 14 August 2019].

[46] C. Dickerson and D. N. Mavris, Architecture and Principles of Systems Engineering, Boca Raton: Taylor and Francis Group, 2009.

[47] A. P. Sage and J. E. Armstrong Jr., Introduction to Systems Engineering, New York: John Wiley & Sons, Inc., 2000.

[48] J. V. Iacobucci, "Rapid Architecture Alternative Modeling: A Framework for Capability-Based Analysis of System of Systems Architectures," Georgia Institute of Technology, Atlanta, 2012.

[49] T. R. Browning, "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and New Directions," *IEEE Transactions on Engineering Management,* vol. 48, no. 3, pp. 292-306, 2001.

[50] M. Filimonov, P. Raju and C. Chapman, "Graph-based modelling of systems interaction in model-based systems engineering environment," in *7th International Systems & Concurrent Engineering for Space Applications Conference*, Madrid, 2016.

[51] N. Markosian, "Physical Object," in *A Companion to Metaphysics*, 2nd ed., J. Kim, E. Sosa and G. S. Rosenkrantz, Eds., Chichester, Wiley-Blackwell, 2009, pp. 486-489.

[52] J. M. Pratt and S. C. Cook, "Using the SoSE Principles Framework," in *27th Annual INCOSE International Symposium*, Adelaide, 2017.

[53] D. D. Walden, G. J. Roedler, K. J. Frosberg, R. D. Hamelin and T. M. Shortell, Eds., "Systems Engineering Overview," in *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 4th ed., Hoboken, John Wiley & Sons, Inc., 2015, pp. 5-24.

[54] International Council on Systems Engineering, "INCOSE System of Systems Primer," July 2018. [Online]. Available: incose.org/products-and-publications/sos-primer. [Accessed 26 August 2019].

[55] C. B. Keating and F. P. Katina, "Systems of systems engineering: prospects and challenges for the emerging field," *Int. J. System of Systems Engineering,* vol. 2, no. 2/3, pp. 234-256, 2011.

[56] K. A. Griendling, "Architect: The Architecture-Based Technology Evaluation and Capability Trade-off Method," Atlanta, 2011.

[57] M. Cantor, "Thoughts on Functional Decomposition," Rational Software, April 2003. [Online]. Available: http://www.therationaledge.com/content/apr_03/f_functionalDecomp_mc.jsp.

[58] K. Sinha, "Structural Complexity and its Implications for the Design of Cyber-Physical Systems," Massachusetts Institute of Technology, Cambridge, 2014.

[59] V. Kim, "Design Space Exploration Method for Identifying Emergent Behavior in Complex Systems," Georgia Institute of Technology, Atlanta, 2016.

[60] J. C. Domerçant, "ARC-VM: An Architecture Real-Options Complexity-Based Valuation Methodology for Military Systems-of-Systems Acquisition," Georgia Institute of Technology, Atlanta, 2011.

[61] J. McEver, "A Complexity Primer for Systems Engineers," November 2015. [Online]. Available: https://www.incose.org/docs/default-source/default-

document-library/complexity-primer-overview.pdf?sfvrsn=91dd81c6_0. [Accessed 17 September 2019].

[62] R. Abbott, "What Makes Complex Systems Complex?," *Journal on Policy and Complex Systems,* vol. 4, no. 2, pp. 77-113, 2018.

[63] M. San Miguel, J. H. Johnson, J. Kertesz, K. Kaski, A. Diaz-Guilera, R. S. MacKay, V. Loreto, P. Erdi and D. Helbing, "Challenges in complex systems science," *Eur. Phys. J. Special Topics,* vol. 214, no. 1, p. 245–271, 2012.

[64] S. A. Sheard and A. Mostashari, "Complexity Types: From Science to Systems Engineering," in *INCOSE International Symposium,* Denver, 2011.

[65] J. Dahmann and G. Roedler, "Moving Towards Standardization for System of Systems Engineering," in *11th System of Systems Engineering Conference (SoSE),* Kongsberg, 2016.

[66] J. J. Johnson IV, A. Tolk and A. Sousa-Poza, "A Theory of Emergence and Entropy in Systems of Systems," *Procedia Computer Science,* vol. 20, p. 283–289, 2013.

[67] A.-M. Grisogono and V. Radenovic, "The Adaptive Stance," *Unifying Themes In Complex Systems,* vol. 8, pp. 714-728, 2011.

[68] J. Dahmann, "System of Systems Pain Points," in *INCOSE International Symposium*, Las Vegas, 2014.

[69] S. Mittal and L. Rainey, "Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering," in *SCSC'15, Summer Simulation Multi-conference*, Chicago, 2015.

[70] J. McEver, S. Sheard, S. Cook, E. Honour, D. Hybertson, J. Krupa, D. McKinney, P. Ondrus, A. Ryan, R. Sheurer, J. Singer, J. Sparber and B. White, "A Complexity Primer for Systems Engineers," International Council on Systems Engineering, San Diego, 2016.

[71] P. Kosky, R. Balmer, W. Keat and G. Wise, "Design Step 3: Evaluation of Alternatives and Selection of a Concept," in *Exploring Engineering*, 3rd ed., Academic Press - Elsevier, 2013, pp. 383-396.

[72] R. Abbott, "The End of (Traditional) Emergence: Introducing Reactive Emergence," *Journal on Policy and Complex Systems,* vol. 2, no. 2, pp. 91-107, 2015.

[73] K. Kitto, "Modelling and Generating Complex Emergent Behavior," Flinders University of South Australia, Adelaide, 2006.

[74] S. J. Rowan, "Descartes, Gödel and Kuhn: Epiphenomenalism Defines a Limit on Reductive Logic," in *Unifying Themes in Complex Systems IX. ICCS 2018. Springer Proceedings in Complexity*, Cham, 2018.

[75] R. B. Laughlin and D. Pines, "The Theory of Everything," *PNAS,* vol. 97, no. 1, pp. 24-31, 2000.

[76] D. G. Green and D. Newth, "Towards a theory of everything? – Grand challenges in complexity and informatics," *Complexity International*, vol. 8, 2001.

[77] R. Abbott, "The Reductionist Blind Spot," *Complexity*, vol. 14, no. 5, pp. 10-22, 2009.

[78] W. E. Herfel, Y. Gao and D. J. Rodrigues, "Chinese Medicine and Complex Systems Dynamics," in *Philosophy of Complex Systems*, Oxford, North Holland - Elsevier, 2011, pp. 675-719.

[79] A. Gawande, The Checklist Manifesto: How to Get Things Right, New York: Metropolitan Books, 2009.

[80] W. Kinsner, "System Complexity and Its Measures: How Complex Is Complex," *Advances in Cogn. Informatics,* vol. 323, p. 265–295, 2010.

[81] S. B. Robinson, "A Modeling Process to Understand Complex System Architectures," Georgia Institute of Technology, Atlanta, 2009.

[82] A. Pugliese, J. Enos and R. Nilchiani, "Acquisition and Development Programs Through the Lens of System Complexity," in *Proceedings of the Fifteenth Annual Acquisition Research Program*, Monterey, 2018.

[83] D. A. Flanigan, "Evaluating the System of Systems Requirements Allocation Process," George Mason University, Fairfax, 2013.

[84] E. Brimhall, R. Wise, R. Simko and J. Huggins, "A Systematic Process for Functional Decomposition in the Absence of Formal Requirements," in *26th Annual INCOSE International Symposium*, Edinburg, 2016.

[85] B. Allen, B. C. Stacey and Y. Bar-Yam, "An Information-Theoretic Formalism for Multiscale Structure in Complex Systems," Cornell University, 16 September 2014. [Online]. Available: arxiv.org/pdf/1409.4708.pdf. [Accessed 23 April 2019].

[86] K. Willcox, D. Allaire, J. Deyst, C. He and G. Sondecker, "Stochastic Process Decision Methods for Complex Cyber-Physical Systems," Massachusetts Institute of Technology, Cambridge, 2011.

[87] J. C. Domerçant, "Information Entropy-Based Complexity Measurement for Systems Engineering and Trade-Off Analysis," in *Proceedings of the 16th Annual Conference on Systems Engineering Research*, Charlottesville, 2018.

[88] F. Jonathan, R. Nilchiani and J. Wade, "Dynamic Complexity Measures for Use in Complexity-Based System Design," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2018-2027, 2017.

[89] S. A. Akundi, "Information Entropy Measures Applied to Hierarchical Complex Technical and Soci-Technical Systems," University of Texas, El Paso, 2016.

[90] S. Lloyd and H. Pagels, "Complexity as Thermodynamic Depth," *Annals of Physics*, vol. 188, no. 1, pp. 186-213, 1988.

[91] C. R. Shalizi, "Methods and Techniques of Complex Systems Science: An Overview," in *Complex Systems Science in Biomedicine*, T. S. Deisboeck and J. Y. Kresh, Eds., Boston, Springer, 2006, pp. 33-114.

[92] N. Ay, E. Olbrich, N. Bertschinger and J. Jost, "A Unifying Framework for Complexity Measures of Finite Systems," Santa Fe Institute, Santa Fe, 2006.

[93] R. S. Kalawsky, "The next generation of grand challenges for systems engineering research," in *Procedia Computer Science, Conference on Systems Engineering Research (CSER'13)*, Atlanta, 2013.

[94] INCOSE, "Systems Engineering Vision 2025," International Council on Systems Engineering, San Diego, 2014.

[95] M. A. Fuentes, "Complexity and the Emergence of Physical Properties," *Entropy*, vol. 16, pp. 4489-4496, 2014.

[96] C. Szabo and Y. M. Teo, "Formalization of Weak Emergence in Multiagent Systems," *ACM Transactions on Modeling and Computer Simulation*, vol. 26, no. 1, pp. 6:1-6:25, 2015.

[97] M. Kokar, S. Singh and S. Lu, "Detection and Learning of Unexpected Behaviors of Systems of Dynamical Systems by Using the Q2 Abstractions," Air Force Research Laboratory, Rome, 2017.

[98] J. Schaffer, "The Metaphysics of Causation (Stanford Encyclopedia of Philosophy)," Stanford University, 5 July 2016. [Online]. Available: https://plato.stanford.edu/entries/causation-metaphysics/. [Accessed 2 October 2019].

[99] P. Menzies, "Counterfactual Theories of Causation (Stanford Encyclopedia of Philosophy)," Stanford University, 10 February 2014. [Online]. Available: https://plato.stanford.edu/entries/causation-counterfactual. [Accessed 2 October 2019].

[100] T. O'Connor and H. Y. Wong, "Emergent Properties (Stanford Encyclopedia of Philosophy)," Stanford University, 3 June 2015. [Online]. Available: https://plato.stanford.edu/entries/properties-emergent/. [Accessed 17 April 2019].

[101] A. Pascual-García, "A Constructive Approach to the Epistemological Problem of Emergence in Complex Systems," *PLoS ONE*, vol. 13, no. 10, 2018.

[102] G. E. Marsh, "The Demystification of Emergent Behavior," 18 July 2013. [Online]. Available: https://arxiv.org/pdf/0907.1117.pdf. [Accessed 26 November 2018].

[103] J. C. Hsu and M. Butterfield, "Modeling Emergent Behavior in Systems-of-Systems," in *INCOSE International Symposium*, San Diego, 2007.

[104] N. A. Baas, "On Higher Structures," 1 December 2015. [Online]. Available: https://arxiv.org/pdf/1509.00403.pdf. [Accessed 19 February 2019].

[105] T. Hofweber, "Logic and Ontology," Stanford Encyclopedia of Philosophy, 11 October 2017. [Online]. Available: https://plato.stanford.edu/entries/logic-ontology. [Accessed 16 April 2021].

[106] S. Ahmed, K. M. Wallace and L. T. Blessing, "Understanding the differences between how novice and experienced designers approach design tasks," *Research in Engineering Design,* vol. 14, pp. 1-11, 2003.

[107] J. S. Hammond, R. L. Keeney and H. Raiffa, "The Hidden Traps in Decision Making," *Harvard Business Review,* no. September-October, pp. 118-126, 1998.

[108] R. W. Batterman, "The Tyranny of Scales," in *The Oxford Handbook of Philosophy of Physics,* Oxford, Oxford University Press, 2013, pp. 255-286.

[109] M. O. Steinhauser, Computational Multiscale Modeling of Fluids and Solids: Theory and Applications, 2nd ed., Berlin: Springer-Verlag, 2017.

[110] P. W. Anderson, "More is Different," *Science,* vol. 177, no. 4047, pp. 393-396, 1972.

[111] A. Ryan, "Emergence is coupled to scope, not level," *Complexity,* vol. 13, no. 2, pp. 67-77, 2007.

[112] G. L. Febres, "A Proposal about the Meaning of Scale, Scope, and Resolution in the Context of the Information Interpretation Process," *Axioms,* vol. 7, no. 11, 2018.

[113] C. Johnson, "What are Emergent Properties and How Do They Affect the Engineering of Complex Systems?," *Reliability Engineering and System Safety,* vol. 91, no. 12, 2006.

[114] A. Kubík, "Toward a Formalization of Emergence," *Artificial Life,* vol. 9, no. 1, pp. 41-65, 2003.

[115] F. Polack and S. Stepney, "Emergent Properties Do Not Refine," *Electronic Notes in Theoretical Computer Science*, vol. 137, pp. 163-181, 2005.

[116] W. C. Wimsatt, "Emergence as Non-Aggregativity and the Biases of Reductionisms," *Foundations of Science,* vol. 5, no. 3, pp. 269-297, 2000.

[117] P. Huneman, "Emergence Made Ontological? Computation versus Combinatorial Approaches," *Philosophy of Science,* vol. 75, p. 595–607, 2008.

[118] L. Jodoin, "Émergence et entropie: Une analyse critique des stratégies explicatives émergentistes basees sur le concept d'entropie," University of Montreal, Montreal, 2015.

[119] J. Goldstein, "Emergence as a Construct: History and Issues," *Emergence,* vol. 1, no. 1, pp. 49-72, 1999.

[120] M. A. Bedau, "Weak Emergence," in *Philosophical Perspectives: Mind, Causation, and World*, J. Tomberlin, Ed., Malden, Blackwell, 1997, pp. 375-399.

[121] D. J. Chalmers, "Strong and Weak Emergence," in *The Re-Emergence of Emergence: The Emergentist Hypothesis From Science to Religion*, P. Davies and P. Clayton, Eds., Oxford, Oxford University Press, 2006.

[122] D. I. Ignatov, "Introduction to Formal Concept Analysis and Its Applications in Information Retrieval and Related Fields," 8 March 2017. [Online]. Available: arxiv.org/pdf/1703.02819.pdf. [Accessed 26 September 2019].

[123] F. Oquendo, "On the Emergent Behavior Oxymoron of System-of-Systems Architecture Description," in *2018 13th Annual Conference on System of Systems Engineering (SoSE)*, Paris, 2018.

[124] E. M. Ronald, M. Sipper and M. S. Capcarrère, "Design, Observation, Surprise! A Test of Emergence," *Artificial Life,* vol. 5, no. 3, pp. 225-239, 1999.

[125] B. McLaughlin and K. Bennett, "Supervenience (Stanford Encyclopedia of Philosophy)," Stanford University, 10 January 2018. [Online]. Available: https://plato. stanford.edu/entries/supervenienc/. [Accessed 1 October 2019].

[126] T. O'Connor, "Emergent Properties," *American Philosophical Quarterly*, vol. 31, pp. 91-104, 1994.

[127] U. Baysan, "Emergence, Function, and Realization," in *Routledge Handbook of Philosophy of Emergence*, S. Gibb, R. F. Hendry and T. Lancaster, Eds., Abingdon, Routledge - Taylor & Francis Group, 2019.

[128] N. A. Baas and C. Emmeche, "On Emergence and Explanation," *Intellectica*, vol. 2, no. 25, pp. 67-83, 1997.

[129] R. Welshon, "Emergence, Supervenience, and Realization," *Philosophical Studies*, vol. 108, no. 1-2, pp. 39-51, 2002.

[130] J. Kim, "Supervenience, Emergence, Realization, Reduction," in *The Oxford Handbook of Metaphysics*, M. J. Loux and D. W. Zimmerman, Eds., Oxford, Oxford University Press, 2005, pp. 556-586.

[131] C. Szabo and Y. M. Teo, "Post-mortem Analysis of Emergent Behavior in Complex Simulation Models," in *Proceedings of the 1st ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*, Montreal, 2013.

[132] C. Szabo and Y. M. Teo, "Semantic Validation of Emergent Properties in Component-Based Simulation Models," in *Ontology, Epistemology, and Teleology for Modeling and Simulation*, vol. 44, A. Tolk, Ed., Berlin, Springer, 2013, p. 319–333.

[133] C.-C. Chen, C. Clack and S. B. Nagl, "A calculus for multi-level emergent behaviours in component-based systems and simulations," in *EPNACS'2007 within ECCS'07*, Dresden, 2007.

[134] M. Á. Fuentes, "Complejidad, emergencia, y cambio teórico," Universidad Nacional de La Plata, La Plata, 2018.

[135] R. Sapolsky, *Emergence and Complexity,* Stanford, CA: YouTube, 2010.

[136] G. Gaeta, "Bifurcation and Symmetry Breaking," *Physics Reports,* vol. 189, no. 1, pp. 1-87, 1990.

[137] P. Grünwald and P. Vitányi, "Shannon Information and Kolmogorov Complexity," 22 July 2010. [Online]. Available: https://homepages.cwi.nl/~paulv/papers/info.pdf. [Accessed 15 October 2019].

[138] T. M. Cover and J. A. Thomas, "The Halting Problem and the Noncomputability of Kolmogorov Complexity," in *Elements of Information Theory*, 2nd ed., New Jersey, John Wiley & Sons, Inc., 2006, pp. 482-484.

[139] W. K. V. Chan, "Interaction Metric of Emergent Behaviors in Agent-Based Simulations," in *Proceedings of the 2011 Winter Simulation Conference*, Phoenix, 2011.

[140] D. Fisch, M. Jänicke, B. Sick and C. Müller-Schloer, "Quantitative Emergence – A Refined Approach Based on Divergence Measures," in *Fourth IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, Budapest, 2010.

[141] P. Hovda, "Quantifying Weak Emergence," *Minds and Machines*, vol. 18, pp. 461-473, 2008.

[142] A. K. Seth, "Measuring emergence via nonlinear Granger causality," *Artificial Life*, vol. 16, no. 2, pp. 179-196, 2010.

[143] R. Dogaru, "Emergence, Locating and Measuring It," in *Systematic Design for Emergence in Cellular Nonlinear Networks: With Applications in Natural Computing and Signal Processing*, Berlin, Springer-Verlag Berlin Heidelberg, 2008, pp. 47-75.

[144] S. D. Mitchell, "Emergence: logical, functional and dynamical," *Synthese*, vol. 185, no. 2, pp. 171-186, 2012.

[145] J. Kim, "Making Sense of Emergence," *Philosophical Studies*, vol. 95, pp. 3-36, 1999.

[146] R. Corry, "Emerging from the causal drain," *Philosophical Studies*, vol. 165, no. 1, pp. 29-47, 2013.

[147] J. C. Hootman and C. Whitcomb, "A Military Effectiveness Analysis and Decision Making Framework for Naval Ship Design and Acquisition," *Naval Engineers Journal*, vol. 117, no. 3, pp. 43-61, 2005.

[148] J. A. Koleser, "Determining surface combatant characteristics requirements through a mission effectiveness analysis framework," Naval Postgraduate School, Monterey, 2007.

[149] V. R. Basili, G. Caldiera and H. D. Rombach, "Goal Question Metric Paradigm," 1994. [Online]. Available: https://www.cs.umd.edu/users/mvz/handouts/gqm.pdf. [Accessed 17 October 2019].

[150] N. Sproles, "Formulating Measures of Effectiveness," *Systems Engineering,* vol. 5, no. 4, pp. 253-263, 2002.

[151] C. Miller, R. S. Carson, S. Fowler, D. J. Gantzer and G. Roedler, "INCOSE Systems Engineering Measurement Primer v2.0," International Council on Systems Eggineering, San Diego, 2010.

[152] A. J. Ryan, "Military Applications of Complex Systems," in *Handbook of the Philosophy of Science: Philosophy of Complex Systems,* vol. 10, C. Hooker, Ed., Oxford, North Holland - Elsevier, 2011, pp. 723-780.

[153] P. K. Davis and D. Blumenthal, "The Base of Sand Problem: A White Paper on the State of Military Combat Modeling," RAND, Santa Monica, 1991.

[154] H. Sayama, Introduction to the Modelng and Analysis of Complex Systems, Geneseo: Open SUNY Textbooks, 2015.

[155] K. A. Richardson, "On the Limits of Bottom-Up Computer Simulation: Towards a Nonlinear Modeling Culture," in *Proceedings of the 36th Hawaii International Conference on System Sciences*, Wailea, 2003.

[156] J. C. Schmidt, "Challenged by Instability and Complexity ... Questioning Classic Stability Assumptions and Presuppositions in Scientific Methodology," in *Handbook of the Philosophy of Science: Philosophy of Complex Systems*, vol. 10, C. Hooker, D. M. Gabbay, P. Thagard and J. Woods, Eds., Oxford, North Holland - Elsevier, 2011, pp. 223-254.

[157] E. W. Weisstein, "Algorithm," From MathWorld--A Wolfram Web Resource, [Online]. Available: http://mathworld.wolfram.com/Algorithm.html. [Accessed 18 October 2019].

[158] A. P. Ershov, "Program," Encyclopedia of Mathematics, [Online]. Available: http://www.encyclopediaofmath.org/index.php?title=Program&oldid=19115. [Accessed 18 October 2019].

[159] E. L. Leiss, A Programmer's Companion to Algorithm Analysis, Boca Raton: Chapman & Hall/CRC - Taylor & Francis Group, 2007.

[160] P. A. Lagerstrom, Matched Asymptotic Expansions: Ideas and Techniques, New York: Springer-Verlag, 1988.

[161] K. J. Cois, W. Pedrycz, R. W. Swiniarski and L. A. Kurgan, Data Mining: A Knowledge Discovery Approach, Springer Science+Business Media, LLC., 2007.

[162] K. Hornik, "Approximation Capabilities of Multilayer Feedforward Networks," *Neural Networks,* vol. 4, pp. 251-257, 1991.

[163] C. Wehmeyer and F. Noé, "Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics," *The Journal of Chemical Physics,* vol. 148, no. 24, pp. 241703:1-9, 2018.

[164] B. Lusch, J. N. Kutz and S. L. Brunton, "Deep learning for universal linear embedding of nonlinear dynamics," *Nature Communications,* vol. 9, 2018.

[165] E. Weisstein, "Game Theory," MathWorld--A Wolfram Web Resource, [Online]. Available: http://mathworld.wolfram.com/GameTheory.html. [Accessed 19 October 2019].

[166] M. Christensen, V. Eyert, A. France-Lanord, C. Freeman, B. Leblanc, A. Mavromaras, S. J. Mumby, D. Reith, D. Rigby, X. Rozanska, H. Schweiger, T.-R. Shan, P. Ungerer, R. Windiks, W. Wolf, M. Yiannourakou and E. Wimmer, "Software Platforms for Electronic/Atomistic/Mesoscopic Modeling: Status and Perspectives," *Integr Mater Manuf Innov,* vol. 6, pp. 92-110, 2017.

[167] E. Weisstein, "Cellular Automaton," MathWorld--A Wolfram Web Resource, [Online]. Available: http://mathworld.wolfram.com/CellularAutomaton.html. [Accessed 19 October 2019].

[168] S. M. Ali and R. M. Zimmer, "Emergence: A Review of the Foundations," *Systems Research and Information Systems,* vol. 8, pp. 1-24, 1998.

[169] Ventana Systems Inc., "What is System Dynamics?," Vensim, 2015. [Online]. Available: https://vensim.com/faqs/what-is-system-dynamics/.

[170] S. E. Phelan, "A Note on the Correspondence Between Complexity and Systems Theory," *Systemic Practice and Action Research,* vol. 12, no. 3, pp. 237-246, 1999.

[171] Y. Zhang, X. Sun, L. Chen, J. Zhang and Y. Liang, "Research on System of Systems Complexity and Decision Making," in *AsiaSim 2012, Part III, CCIS 325*, Shanghai, 2012.

[172] K. Beven, "Facets of uncertainty: epistemic uncertainty, non-stationarity, likelihood, hypothesis testing, and communication," *Hydrological Sciences Journal,* vol. 61, no. 9, pp. 1652-1665, 2016.

[173] D. Goldberg, "What Every Computer Scientist Should Know about Floating-Point Arithmetic," *ACM Computing Surveys,* vol. 21, no. 1, pp. 5-44, 1991.

[174] M. Squair, "Epistemic, ontological and aleatory risk," Critical Uncertainties, 10 November 2009. [Online]. Available: https://criticaluncertainties.com/2009/10/11/epistemic-and-aleatory-risk/. [Accessed 1 May 2019].

[175] D. N. Talley, "Methodology for the Conceptual Design of a Robust and Opportunistic System-of-Systems," Georgia Institute of Technology, Atlanta, 2008.

[176] P. Meyer, "Evaluating the Effects of Model Simplifications on the Transference of Policies Learned in Simulation," Georgia Institute of Technology, Atlanta, 2020 (forthcoming).

[177] S. Rangavajhala, C. Liang and S. Mahadevan, "Inclusion of Aleatory and Epistemic Uncertainty in Design Optimizatin," in *10th World Congress on Structural and Multidisciplinary Optimization*, Orlando, 2013.

[178] K. S. Bhachu, R. T. Haftka and N. H. Kim, "Quantifying Effect of Uncertainty Sources on Optimum in Tolerance Optimization," in *54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Boston, 2013.

[179] J. Fender, L. Graff, H. Harbrecht and M. Zimmerman, "Identifying Key Parameters for Design Improvement in High-Dimensional Systems with Uncertainty," *Journal of Mechanical Design*, vol. 136, 2014.

[180] J. Fromm, "Ten Questions about Emergence," 2005. [Online]. Available: https://arxiv.org/ftp/nlin/papers/0509/0509049.pdf. [Accessed 16 November 2018].

[181] J. M. Epstein, "Agent-Based Computational Models And Generative Social Science," *Complexity*, vol. 4, no. 5, pp. 41-60, 1999.

[182] B. Baker, I. Kanitscheider, T. Markov, Y. Wu, G. Powell, B. McGrew and I. Mordatch, "Emergent Tool Use From Multi-Agent Autocurricula," OpenAI, 17 September 2019. [Online]. Available: https://arxiv.org/pdf/1909.07528.pdf. [Accessed 19 October 2019].

[183] T. Moncion, P. Amar and G. Hutzler, "Automatic characterization of emergent phenomena in complex systems," *Journal of Biological Physics and Chemistry*, vol. 10, pp. 16-23, 2010.

[184] J. R. Boyd, "Aerial Attack Study," 1964.

[185] F. C. Blesse, No Guts No Glory!, 1955.

[186] J. R. Boyd, A Discourse on Winning and Losing, G. T. Hammond, Ed., Maxwell AFB, Alabama: Air University Press, 2018.

[187] I. T. Brown, A New Conception of War: John Boyd, the US Marines, and a New Conception of Warfare, Quantico: Marine Corps University Press, 2018.

[188] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model," in *SIGGRAPH '87 Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, Anaheim, 1987.

[189] U. Wilensky, "NetLogo Flocking Model," Center for Connected Learning and Computer-Based Modeling, Northwestern University, 1998. [Online]. Available: http://ccl.northwestern.edu/netlogo/models/Flocking. [Accessed 20 October 2019].

[190] J. L. Choate, "Extending AFSIM with Behavioral Emergence," Air Force Institute of Technology, Wright-Patterson Air Force Base, 2017.

[191] G. Minati and I. Licata, "Emergence as Mesoscopic Coherence," *Systems,* vol. 1, pp. 50-65, 2013.

[192] R. L. Cooke, "Almost-Periodic Functions," *The American Mathematical Monthly,* vol. 88, no. 7, pp. 515-526, 1981.

[193] A. I. Stepanets, "Classification of Periodic Functions and the Rate of Convergence of their Fourier Series," *Mathematics of the USSR-Izvestiya,* vol. 28, no. 1, pp. 99-132, 1987.

[194] D. C. Dennett, "Real Patterns," *The Journal of Philosophy,* vol. 88, no. 1, pp. 27-51, 1991.

[195] G. Samaey, T. Holvoet and T. De Wolf, "Using Equation-Free Macroscopic Analysis for Studying Self-Organising Emergent Solutions," in *Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, Venice, 2008.

[196] N. Ay, R. Der and M. Prokopenko, "Guided self-organization: perception–action loops of embodied systems," *Theory in Biosciences,* vol. 131, no. 3, pp. 125-127, 2012.

[197] J. D. Halley and D. A. Winkler, "Classification of Emergence and its Relation to Self-Organization," *Complexity,* vol. 13, no. 5, pp. 10-15, 2008.

[198] W. Hofkirchner and M. Schafranek, "General System Theory," in *Philosophy of Complex Systems*, Oxford, North Holland - Elsevier, 2011, pp. 177-194.

[199] J. Crutchfield, "The Calculi of Emergence: Computation, Dynamics, and Induction," in *Proceedings of the Oji International Seminar: Complex Systems - from Complex Dynamics to Artificial Reality*, Numazu, 1993.

[200] E. Bonabeau, J.-L. Dessalles and A. Grumbach, "Characterizing Emergent Phenomena (1): A Critical Review," *Revue Internationale de Systémique,* vol. 9, no. 3, pp. 327-346, 1995.

[201] N. W. Watkins, G. Pruessner, S. C. Chapman, N. B. Crosby and H. J. Jensen, "25 Years of Self-Organized Criticality: Concepts and Controversies," *Space Science Reviews,* vol. 198, no. 1-4, pp. 3-44, 2016.

[202] N. Fernández, C. Maldonado and C. Gershenson, "Information Measures of Complexity, Emergence, Self-organization, Homeostasis, and Autopoiesis," 1 July 2013. [Online]. Available: http://arxiv.org/pdf/1304.1842.pdf. [Accessed 25 October 2019].

[203] A. Mathieu, P.-H. Cournède, D. Barthélémy and P. De Reffye, "Rythms and Alternating Patterns in Plants as Emergent Properties of a Model of Interaction between Development and Functioning," *Annals of Botany*, vol. 101, pp. 1233-1242, 2008.

[204] C. N. El-Hani and S. Pihlström, "Emergence Theories and Pragmatic Realism," *Essays in Philosophy*, vol. 3, no. 2, 2002.

[205] C. Truesdell, "A Precise Upper Limit for the Correctness of the Navier-Stokes Theory with Respect to the Kinetic Theory," *Journal of Statistical Physics*, vol. 1, no. 2, pp. 313-318, 1969.

[206] M. Prokopenko, F. Boschetti and A. J. Ryan, "A Information-Theoretic Primer on Complexity, Self-Organization, and Emergence," *Complexity*, vol. 15, no. 1, pp. 11-28, 2008.

[207] W. A. Wright, R. E. Smith, M. Danek and P. Greenway, "A Generalisable Measure of Self-Organisation and Emergence," in *International Conference on Artificial Neural Networks: ICANN 2001*, Vienna, 2001.

[208] I. Licata and G. Minati, "Emergence, Computation and the Freedom Degree Loss Information Principle in Complex Systems," *Foundations of Science*, vol. 22, no. 4, pp. 863-881, 2017.

[209] M. Wilkerson-Jerde, F. Stonedahl and U. Wilensky, "Flocking Vee Formations model," Center for Connected Learning and Computer-Based Modeling, Northwestern University, 2009. [Online]. Available:

http://ccl.northwestern.edu/netlogo/models/FlockingVeeFormations. [Accessed 21 October 2019].

[210] J. A. Smith, "Numerical and Analytical Approaches to Modelling 2D Flocking Phenomena," University of Melbourne, Melbourne, 2011.

[211] J. Winning and W. Betchel, "Being Emergence vs. Pattern Emergence: Complexity, control, and goal-directedness in biological systems," in *Routledge Handbook of Emergence*, S. C. Gibb, R. F. Hendry and T. Lancaster, Eds., London, Routledge, 2019, pp. 134-144.

[212] R. Abbott, "The reductionist blind spot: downward reduction and upward conceptualization," 20 May 2014. [Online]. Available: https://www.researchgate.net/publication/228680647_The_reductionist_blind_spot_downward_reduction_and_upward_conceptualization. [Accessed 11 October 2019].

[213] J. Kim, Physicalism, or Something near Enough, Oxford: Princeton University Press, 2005.

[214] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, Chichester: Wiley, 2009.

[215] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca and V. G. da Fonseca, "Performance Assessment of Multiobjective Optimizers: An Analysis and Review," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 117-32, 2003.

[216] J. Teich, "Pareto-Front Exploration with Uncertain Objectives," in *Evolutionary Multi-Criterion Optimization, First International Conference*, Berlin, 2001.

[217] K. Brown, "Von Neumann's Postulate and Bell's Freedom," MathPages, 2020. [Online]. Available: https://mathpages.com/rr/s9-06/9-06.htm. [Accessed 7 January 2021].

[218] G. Hassan-Coring, "Inferring Causality in Agent-Based Simulations - Literature Review," December 2018. [Online]. Available: https://arxiv.org/pdf/1901.04836.pdf. [Accessed 19 February 2019].

[219] A. K. Seth, "Measuring autonomy and emergence via Granger causality," *Artificial Life*, vol. 16, no. 2, pp. 179-196, 2010.

[220] A. Kathpalia and N. Nagaraj, "Causality Testing: A Data Compression Framework," 18 February 2018. [Online]. Available: https://arxiv.org/pdf/1710.04538.pdf. [Accessed 5 March 2019].

[221] E. O'Toole, V. Nallur and S. Clarke, "Towards Decentralised Detection of Emergence in Complex Adaptive Systems," in *IEEE Eighth International Conference on Self-Adaptive and Self-Organizing Systems*, London, 2014.

[222] R. Ouyang, S. Curtarolo, E. Ahmetcik, M. Scheffler and L. M. Ghiringhelli, "SISSO: a compressed-sensing method for identifying the best low-dimensional descriptor in an immensity of offered candidates," 27 June 2018. [Online]. Available: https://arxiv.org/pdf/1710.03319.pdf. [Accessed 11 July 2018].

[223] C. M. Acosta, R. Ouyang, A. Fazzio, M. Scheffler, L. M. Ghiringhelli and C. Carbogno, "Analysis of Topological Transitions in Two-Dimensional Materials by Compressed Sensing," 28 May 2018. [Online]. Available: https://arxiv.org/pdf/1805.10950.pdf. [Accessed 2 October 2019].

[224] S. L. Brunton, J. L. Proctor and J. N. Kutz, "Discovering governing equations from data by sparse identification on nonlinear dynamical systems," *PNAS,* vol. 113, no. 15, pp. 3932-3937, 2016.

[225] S. L. Brunton and J. N. Kutz, "Methods for data-driven multiscale model discovery for materials," *Journal of Physics: Materials*, vol. 2, no. 4, 2019.

[226] A. Griewank and A. Walther, Evaluating Derivatives - Principles and Techniques of Algorithmic Differentiation, 2nd ed., Philadelphia: Society for Industrial and Applied Mathematics, 2008.

[227] J. Borwein and P. Borwein, "On the Complexity of Familiar Functions and Numbers," *SIAM Review*, vol. 30, no. 4, pp. 589-601, 1988.

[228] "The Running Time of Programs," in *Foundations of Computer Science - C Edition*, Computer Science Press, 1992, pp. 89-155.

[229] K. Rutanen, "O-notation in Algorithm Analysis (updated)," Cornell University, 4 January 2020. [Online]. Available: https://arxiv.org/pdf/1309.3210.pdf. [Accessed 2 November 2020].

[230] J. M. Borwein and P. B. Borwein, "Chapter 6: The Complexity of Algebraic Functions," in *Pi and the AGM: A Study in Analytic Number Theory and Computational Complexity*, New York, John Wiley & Sons, Inc., 1987, pp. 200-218.

[231] J. M. Borwein and P. B. Borwein, "Chapter 7: Algorithms for Elementary Functions," in *Pi and the AGM: A Study in Analytic Number Theory and Computational Complexity*, New York, John Wiley & Sons, Inc., 1987, pp. 219-229.

[232] A. Schönhage, A. F. W. Grotefeld and E. Vetter, Fast Algorithms: A Multitape Turing Machine Implementation, Mannheim: Bibliographishes Institut & F.A. Brockhaus AG, 1994.

[233] E. Jeřábek and (https://mathoverflow.net/users/12705/emil-je%c5%99%c3%a1bek), "Computational complexity of calculating the nth root of a real number," MathOverflow, 01 08 2014. [Online]. Available: https://mathoverflow.net/q/99445. [Accessed 09 10 2020].

[234] D. Harvey and J. van der Hoeven, "Integer multiplication in time O(n log n)," *HAL*, pp. 1-42, 2019.

[235] D. E. Knuth, "Arithmetic," in *The Art of Computer Programming: Seminumerical Algorithms*, 3rd ed., vol. 2, Reading, Addison-Wesley, 2014.

[236] A. Montanaro, "Computational Complexity - Lecture Notes Part III," 28 May 2013. [Online]. Available: https://people.maths.bris.ac.uk/~csexam/teaching/cc-lecturenotes.pdf. [Accessed 2 December 2020].

[237] S. Alaee, K. Kamgar and E. Keogh, "Matrix Profile XXII: Exat Discovery of Time Series Motifs and DTW," arXiv, 16 September 2020. [Online]. Available: https://arxiv.org/abs/2009.07907. [Accessed 4 December 2020].

[238] "What is the Matrix Profile?," Matrix Profile Foundation, 2020. [Online]. Available: https://matrixprofile.org. [Accessed 4 December 2020].

[239] M. Ziatdinov, O. Dyck, A. Maksov, X. Li, X. Sang, K. Xiao, R. R. Unocic, R. Vasudevan, S. Jesse and S. V. Kalinin, "Deep Learning of Atomically Resolved Scanning Transmission Electron Microscopy Images: Chemical Identification and Tracking Local Transformations," *ACS Publications*, vol. 11, no. 12, pp. 12742-12752, 2017.

[240] S. V. Kalinin, S. Zhang, M. Valleti, H. Pyles, D. Baker, J. J. De Yoreo and M. Ziatdinov, "Exploring particle dynamics during self-organization processes via rotationally invariant latent representations," 2 September 2020. [Online]. Available: https://arxiv.org/abs/2009.00783. [Accessed 13 December 2020].

[241] MathWorks, "fft," The MathWorks, Inc., 2020. [Online]. Available: https://www.mathworks.com/help/matlab/ref/fft.html. [Accessed 11 October 2020].

[242] N. Wagner and Z. Michalewicz, "Adaptive and Self-adaptive Techniques for Evolutionary Forecasting Applications Set in Dynamic and Uncertain Environments," in *Foundations of Computational Intelligence: Bio-Inspired Data*

Mining, vol. 4, A. Abraham, A. Hassanien and A. P. d. L. F. de Carvalho, Eds., Berlin, Springer-Verlag, 2009, pp. 3-21.

[243] K. Webb, "Section 10: Fouier Analysis - MAE 4020/5020 - Numerical Methods with MATLAB," 2010. [Online]. Available: http://web.engr.oregonstate.edu/~webbky/MAE4020_5020_files/Section%2010%20Fourier%Analysis.pdf. [Accessed 6 December 2020].

[244] Center for Connected Learning and Computer-Based Modeling, "NetLogo 6.1.1 User Manual - Programming Guide," Northwestern University, 26 September 2019. [Online]. Available: https://ccl.northwestern.edu/netlogo/docs/programming.html. [Accessed 1 September 2019].

[245] M. A. Cummings, "Identifying and Quantifying Emergent Behavior through System of Systems Modeling and Simulation," Naval Post Graduate School, Monterey, 2015.

[246] D. G. Cacuci, Sensitivity and Uncertainty Analysis - Theory, vol. I, Boca Raton: Chapman & Hall/CRC, 2003.

[247] S. L. Brunton and J. N. Kutz, Data Driven Science & Engineering: Machine Learning, Dynamical Systems, and Control, Cambridge: Cambridge University Press, 2017.

[248] Center for Connected Learning and Computer-Based Modeling, "NetLogo 6.1.1 User Manual - Dictionary," 26 September 2019. [Online]. Available: https://ccl.northwestern.edu/netlogo/docs/dictionary.html. [Accessed 10 March 2020].

[249] The MathWorks, Inc., "polyfit - Polynomial curve fitting," The MathWorks, Inc., 2018. [Online]. Available: https://www.mathworks.com/help/matlab/ref/polyfit.html. [Accessed 15 December 2018].

[250] Unknown, "The Battle of Midway June 3-6 1942 - Appendix Fourteen: United States Navy Fighter Tactics," Internet Archive Wayback Machine, 4 February 2012. [Online]. Available: https://web.archive.org/web/20120204195154/http://www.centuryinter.net/midway/appendix/appendixfourteen_usvftac.html. [Accessed 30 May 2019].

[251] R. T. Jantzen, "Geodesics on the Torus and other Surfaces of Revolution Clarified Using Undergraduate Physics Tricks with Bonus: Nonrelativistic and Relativistic

Kepler Problems," 23 May 2010. [Online]. Available: https://arxiv.org/pdf/1212.6206.pdf. [Accessed 11 February 2021].

[252] M. L. Irons, "The Curvature and Geodesics of the Torus," 17 November 2005. [Online]. Available: www.rdrop.com/~half/math/torus/torus.geodesics.pdf. [Accessed 11 February 2021].

[253] J. G, "What is formula for the distance between two points on a topological torus?," yahoo!answers, 2011. [Online]. Available: https://answers.yahoo.com/. [Accessed 11 February 2021].

[254] E. Soria, F. Schiano and D. Floreano, "SwarmLab: a Matlab Drone Swarm Simulator," 23 September 2020. [Online]. Available: https://arxiv.org/pdf/2005.02769v2.pdf. [Accessed 14 October 2020].

[255] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben and T. Vicsek, "Optimized flocking of autonomous drones in confined environments," *Science Robotics,* vol. 3, no. 20, pp. 1-13, 2018.

[256] R. Olfati-Saber and R. M. Murray, "Distributed Cooperative Control of Multiple Vehicle Formations Using Structural Potential Functions," in *IFAC - 15th Triennial World Congress,* Barcelona, 2002.

[257] A. Bottinelli, D. T. Sumpter and J. L. Silverberg, "Emergent Structural Mechanisms for High-Density Collective Motion Inspired by Human Crowds," *Phsycial Review Letters,* vol. 117, no. 22-25, p. 228301, 2016.

[258] D. Helbing, I. Farkas and T. Viscek, "Simulating dynamical features of escape panic," *Nature,* vol. 407, pp. 487-490, 2000.

[259] J. C. Hsu, E. Axelband, A. M. Madni, C. Dagli and D. McKinney, "Can We Engineer the Emergent Behavior of A System-of-Systems?," in *INCOSE International Symposium,* Singapore, 2009.

[260] T. Ufuk, N. Bliss, N. Cooke, M. L. Cummings, A. Llorens, H. Shrobe and L. Zuck, "Assured Autonomy: Path Toward Living With Autonomous Systems We Can Trust," 27 October 2020. [Online]. Available: https://arxiv.org/abs/2010.14443. [Accessed 16 April 2021].

[261] L. Gammon, J. Tang, J. Fan and E. Kee, "TRIDENT: Time-variant Response Investigator for Determination of Effective Naval-Task Management," in *26th Annual External Advisory Board Review*, Atlanta, 2018.

[262] D. H. Lamb, "The effects of two stressors on state anxiety for students who differ in trait anxiety," *Journal of Research in Personality,* vol. 7, no. 2, pp. 116-126, 1973.

[263] M. R. Leary, M. L. Terry, A. B. Allen and E. B. Tate, "The Concept of Ego Threat in Social and Personality Psychology: Is Ego Threat a Viable Scientific Construct?," *Personality and Social Psychology Review,* vol. 13, no. 3, 2009.

[264] I. G. Kevrekidis and G. Samaey, "Equation-Free Multiscale Computation: Algorithms and Applications," *Annual Review of Physical Chemistry,* vol. 60, pp. 321-344, 2009.

[265] Y. Kevrekidis and G. Samaey, "Equation-Free Modeling," *Scholarpedia,* vol. 5, no. 9, pp. 4847, revision 137023, 2010.

[266] "Conway's Game of Life," Wikipedia, 27 October 2020. [Online]. Available: https://en.wikipedia.org/wiki/Conway%27s_Game_of_Life. [Accessed 16 November 2020].

[267] P. Rendell, "A Turing Machine in Conway's Game of Life, extendable to a Universal Turing Machine," 14 September 2015. [Online]. Available: http://rendell-attic.org/gol/tm.htm. [Accessed 16 November 2020].

[268] P. Rendell, "Turing Machine Universality of the Game of Life," University of the West of England, Bristol, 2014.

[269] P. Rendell, "This is a Universal Turing Machine (UTM) implemented in Conway's Game of Life," 23 March 2011. [Online]. Available: http://rendell-attic.org/gol/utm/index.htm. [Accessed 25 November 2020].

[270] F. Martinelli, "Reproducing sinusoid with autoregressive discrete model," Cross Validated, 27 November 2018. [Online]. Available: https://stats.stackexchange.com/q/378996. [Accessed 15 December 2020].

[271] ritkivmath, "Time Series Talk: Stationarity," YouTube, 8 July 2019. [Online]. Available: https://youtu.be/oY-j2Wof51c.

[272] R. H. Shumway and D. S. Stoffer, Time Series Analysis and Its Applications: With R Examples, 4th ed., Berlin: Springer, 2016.

[273] A. Seth, "Granger Causality," Scholarpedia, 23 August 2012. [Online]. Available: www.scholarpedia.org/article/Granger_causality. [Accessed 24 October 2019].

[274] D. Giles, "Econometrics Beat: Dave Giles' Blog," Blogspot, 29 April 2011. [Online]. Available: https://davegiles.blogspot.com/2011/04/testing-for-granger-causality.html. [Accessed 1 May 2020].

[275] P. J. Brockwell, Introduction to Time Series and Forecasting, 3rd ed., Springer International Publishing, 2016.

[276] R. H. Shumway and D. S. Stoffer, Time Series Analysis and Its Applications: With R Examples, 4th ed., Springer International Publishing, 2017.

[277] A. Papana, C. Kyrtsou, D. Kugiumtzis and C. Diks, "Identifying causal relationships in case of non-stationary time series," Universiteit van Amsterdam, Center for Nonlinear Dynamics in Economics and Finance, Amsterdam, 2014.

[278] P. Perron, M. Shintani and T. Yabu, "Testing for Flexible Nonlinear Trends with an Integrated or Stationary Noise Component," *Oxford Bulletin of Economics and Statistics,* vol. 79, no. 5, pp. 822-850, 2017.

[279] A. Seth, "Granger Causality," *Scholarpedia,* vol. 2, no. 7, pp. 1667., revision #127333, 2007.

[280] F. A. Razak and H. J. Jensen, "Quantifying 'Causality' in Complex Systems: Understanding Transfer Entropy," *PLoS ONE,* vol. 9, no. 6, p. e99462, 2014.

[281] J. D. Sterman, "System dynamics at sixty: the path forward," *System Dynamics Review,* vol. 34, no. 1-2, pp. 5-47, 2018.

[282] J. W. Forrester, "System dynamics, systems thinking, and soft OR," *System Dynamics Review,* vol. 10, no. 2-3, pp. 245-256, 1994.

[283] M. A. Hernán and J. M. Robins, Causal Inference: What If, Boca Raton: CRS Press/Taylor & Francis Group, 2020.

[284] D. T. Kaplan, "Statistical Modeling: A Fresh Approach," 2017. [Online]. Available: dtkaplan.github.io/SM2-bookdown/preface-to-this-electronic-version.html. [Accessed 14 December 2020].

[285] J. Pearl, Causality: Models, Reasoning, and Inference, New York: Cambridge University Press, 2009.

[286] R. B. Kline, Principles and Practice of Structural Equation Modeling (Methodology in the Social Sciences), New York: The Guilford Press, 2015.

[287] J. M. Bishop, "Artificial Intelligence is stupid and causal reasoning won't fix it," 20 July 2020. [Online]. Available: https://arxiv.org/abs/2008.07371.

[288] Microsoft, "dowhy," GitHub, 2020. [Online]. Available: https://github.com/microsoft/dowhy/. [Accessed 18 December 2020].

[289] A. Sharma and E. Kiciman, "DoWhy - A library for causal inference," Microsoft, 21 August 2018. [Online]. Available: https://microsoft.com/en-us/research/blog/dowhy-a-library-for-causal-inference/. [Accessed 18 December 2020].

[290] A. Molenaar and D. Yeaw, "Gaphor: UML/SysML Modeling," Gaphor, 2020. [Online]. Available: https://gaphor.org. [Accessed 19 December 2020].

[291] B. Avent, A. Ritz, T. M. Murali, J. Cadena and Y. Keneshloo, "halp: Hypergraph Algorithms Package," 8 October 2014. [Online]. Available: https://murali-group.github.io/halp/. [Accessed 22 October 2020].

[292] A. Antelmi, G. Cordasco, B. Kamiński, P. Prałat, V. Scarano, C. Spagnuolo and P. Szufel, "Analyzing, Exploring, and Visualizing Complex Networks via Hypergraphs using SimpleHypergraphs.jl," *Internet Mathematics*, vol. 1, no. 1, pp. 1-32, 2020.

[293] D. Ranathunga, H. Nguyen and M. Roughan, "MGtoolkit: A python package for implementing metagraphs," *SoftwareX*, vol. 6, pp. 91-93, 2017.

[294] The Sage Development Team, "Sage 9.1 Reference manual: Graph Theory - Hypergraph generators," [Online]. Available: https://doc.sagemath.org/html/en/reference/graphs/sage/hypergraph_generators.html. [Accessed 22 October 2020].

[295] B. Praggastis, D. Arendt, E. Purvine, C. Joslyn, M. Raugas, S. Aksoy and K. Monson, "HyperNetX," 1 May 2019. [Online]. Available: https://github.com/pnnl/HyperNetX. [Accessed 22 October 2020].

[296] P. W. Battaglia and et al., "Relational inductive biases, deep learning, and graph networks," 17 October 2018. [Online]. Available: https://arxiv.org/abs/1806.01261v3.

[297] Google Deepmind, "Graph Nets," 2020. [Online]. Available: https://github.com/deepmind/graph_nets. [Accessed 19 December 2020].

[298] K. Cranmer, J. Brehmer and G. Louppe, "The frontier of simulation-based inference," *PNAS,* vol. 117, no. 48, pp. 30055-30062, 2020.

[299] System Dynamics Society, "Useful Open Source Tools," System Dynamics Society, 2020. [Online]. Available: https://systemdynamics.org/tools/useful-open-source-tools/.

[300] Team SimPy, "SimPy - Discrete event simulation for Python," 2020. [Online]. Available: https://simpy.readthedocs.io/en/latest. [Accessed 19 December 2020].

[301] J. Willard, X. Jia, M. Steinbach and V. Kumar, "Integrating Physics-Based Modeling with Machine Learning: A Survey," 10 March 2020. [Online]. Available: https://arxiv.org/abs/2003.04919.

[302] M. Cranmer, A. Sanchez-Gonzalez, P. Battaglia, R. Xu, K. Cranmer, D. Spergel and S. Ho, "Discovering Symbolic Models from Deep Learning with Inductive

Biases," in *34th Conference on Neural Information Processing Systems*, Vancouver, 2020.

[303] X.-Y. Zhang, C.-L. Liu and C. Y. Suen, "Towards Robust Pattern Recognition: A Review," *Proceedings of the IEEE,* vol. 108, no. 6, pp. 894-922, 2020.

[304] S. Asht and R. Dass, "Pattern Recognition Techniques: A Review," *International Journal of Computer Science and Telecommunications,* vol. 3, no. 8, pp. 25-29, 2012.

[305] S. K. Pal and P. Mitra, Pattern Recognition Algorithms for Data Mining, Boca Raton: Chapman & Hall/CRC, 2004.

[306] M. Hause, "How to Fail at MBSE," INCOSE (Atego Presentation), May 2013. [Online]. Available: https://www.incose.org/docs/default-source/texas-gulf-coast/m_hause_how_to_fail_at_mbse.pdf.

[307] R. H. Myers, D. C. Montgomery and C. M. Anderson-Cook, Response Surface Methodology: Process and Product Optimization Using Designed Experiments, 3rd ed., New Jersey: John Wiley & Sons, Inc., 2009.

[308] L. Fausett, Fundamentals of Neural Networks: Architectures, Algorithms, and Applications, B. Baker, Ed., Upper Saddle River, NJ: Prentice-Hall Inc., 1994.

[309] dynamicslab, "PySINDy," 19 June 2020. [Online]. Available: https://github.com/dynamicslab/pysindy.

[310] S.-M. Udrescu and M. Tegmark, "Al Feynman: A physics-inspired method for symbolic regression," *Science Advances,* vol. 6, no. eaay2631, pp. 1-16, 2020.

[311] S.-M. Urudescu, A. Tan, J. Feng, O. Neto, T. Wu and M. Tegmark, "AI Feynman 2.0: Pareto-optimal symbolic regression exploiting graph modularity," 18 June 2020. [Online]. Available: arxiv.org/pdf/2006.10782. [Accessed 7 December 2020].

[312] Z. Liu and M. Tegmark, "AI Poincare: Machine Learning Conservation Laws from Trajectories," 9 November 2020. [Online]. Available: https://arxiv.org/abs/2011.04698.

[313] M. Schmidt and H. Lipson, "Distilling Free-Form Natural Laws from Experimental Data," *Science*, vol. 324, pp. 81-85, 2009.

[314] J. Collier, "A Dynamical Account of Emergence," *Cybernetics and Human Knowing,* vol. 15, no. 3-4, pp. 75-86, 2008.

[315] R. Moyal, T. Fekete and S. Edelman, "Dynamical Emergence Theory (DET): A Computational Account of Phenomenal Consciousness," *Minds and Machines,* vol. 30, pp. 1-21, 2020.

[316] T. M. Thomas II, "Army Capability Development Needs to Go Back to the Future," U.S. Army War College, Carlisle, 2016.

[317] J. R. Fox, Defense Acquisition Reform, 1960–2009: An Elusive Goal, Washington D.C.: United States Army, 2011.

[318] T. Hone, "Fixing Navy Acquisition," U.S. Naval Institute Blog, 29 April 2016. [Online]. Available: https://blog.usni.org/posts/2018/10/09/navy-should-adopt-startup-culture-but-avoid-the-pitfalls. [Accessed 10 October 2018].

[319] C. Goddard, H. Fireman and C. Deegan, "A Question of Cost," *Armed Forces Journal,* 2007.

[320] M. Sullivan, "Best Practices: Stronger Practices Needed to Improve DoD Technology Transition Process," United States Government Accountability Office, Washington D.C., 2006.

[321] M. Mackin, D. Moldafsky, P. Almoguera, J. L. Beier, L. Ettaro, L. Greifner, K. Gurka, K. Hassinger, K. Lenane, J. C. Madar, R. Sun and C. J. Yun, "Need to

Address Fundamental Weaknesses in LCS and Frigate Acquisition Strategies," United States Government Accountability Office, Washington D.C., 2016.

[322] M. H. Spiknall and M. Wade, "Reducing the Construction Contract Cycle for Naval Auxiliary Ships," in *Proceedings of the Ship Production Symposium*, New Orleans, 1992.

[323] National Academy of Sciences, Pre-Milestone A and Early-Phase Systems Engineering: A Retrospective Review and Benefits for Future Air Force Systems Acquisition, Washington D.C.: The National Academies Press, 2008.

[324] N. C. L. Knisely, "Cyber-Physical Acquisition Strategy for COTS-Based Agility-Driven Engineering," Georgia Institute of Technology, Atlanta, 2016.

[325] B. Keenan, "Capabilities Development System Acquisition Management - Executive Primer," Army Force Management School, Fort Belvoir, 2013.

[326] Joint Staff, "Operation of the Defense Acquisition System," Department of Defense, Washington D.C., 2017.

[327] M. L. Cavanaugh, "It's Time to End the Tyranny of Ends, Ways, and Means," Modern War Institute at West Point, 24 July 2017. [Online]. Available:

https://mwi.usma.edu/time-end-tyranny-ends-ways-means/. [Accessed 19 July 2019].

[328] J. W. Meiser, "Are Our Strategic Models Flawed?: Ends + Ways + Means = (Bad) Strategy," *Parameters*, vol. 46, no. 4, pp. 81-91, 2017.

[329] Oxford Dictionaries, "Definition of capability in English," Lexico.com, 19 July 2019. [Online]. Available: https://www.lexico.com/en/definition/capability. [Accessed 19 July 2019].

[330] Joint Chiefs of Staff, Capabilities-Based Assessment (CBA) User's Guide Version 3, Washington D.C.: Department of Defense, 2009.

[331] R. Martens and M. Rempel, "High-Level Methodologies to Evaluate Naval Task Groups," *Naval Engineers Journal,* vol. 123, no. 4, pp. 67-80, 2011.

[332] M. W. P. Tackett, C. A. Mattson and S. M. Ferguson, "A Model for Quantifying System Evolvability Based on Excess and Capacity," *Journal of Mechanical Design,* vol. 136, no. 5, 2014.

[333] N. H. Guertin, D. C. Schmidt and W. Scherlis, "Capability Composition and Data Interoperability to Achieve More Effective Results Than DoD System-of-Systems

Strategies," in *Proceedings of the Fifteenth Annual Acquisition Research Symposium - Volume I*, Monterey, 2018.

[334] L. D. Ferreiro, "Comparing Ship Versus Aircraft Development Costs," in *The Fifteenth Annual Acquisition Research Symposium*, Monterey, 2018.

[335] G. A. Davis, M. L. Giles and D. M. Tate, "Complexity in an Unexpected Place: Quantities in Selected Acquisition Reports," in *Proceedings of the Fifteenth Annual Acquisition Research Symposium - Volume I*, Monterey, 2018.

[336] I. Lewis, "Persistent Platforms -- The DDG 51 Case," Naval Postgraduate School, Monterey, 2015.

[337] R. G. Carlucci, "Analysis of Alternatives (AoA) Process Improvement Study," Center for Army Analysis, Fort Belvoir, 2016.

[338] B. Brown, Introduction to Defense Acquisition Management, Fort Belvoir: Defense Acquisition University, 2010.

[339] J. C. Mun and T. J. Housel, "Flexible and Adaptable Ship Options: Assessing the Future Value of Incorporating Flexible Ships Design Features Into New Navy Ship Concepts," in *Proceedings of the Fourteenth Annual Acquisition Symposium*, Monterey, 2017.

[340] N. Doerry, "Institutionalizing Modular Adaptable Ship Technologies," 7 July 2011. [Online]. Available: http://doerry.org/norbert/papers/20110707ModularAdaptableShip-final.pdf. [Accessed 4 February 2018].

[341] C. D. Connors, "Agent-based Modeling Methodology for Analyzing Weapons Systems," Air Force Institute of Technology, Wright-Patterson Air Force Base, 2015.

[342] J. Meyer, "A Risk-Based Approach to Optimal Margins in Ship Design," Massachusetts Institute of Technology, Cambridge, 2002.

[343] B. Kassel, S. Cooper and A. Mackenna, "Rebuilding the NAVSEA Early Stage Ship Design Environment," in *ASNE Day - Engineering the Affordable Global Navy Through Innovation*, Arlington, 2010.

[344] T. McKenney, L. F. Kemink and D. J. Singer, "Adapting to Changes in Design Requirements Using Set-Based Design," *Naval Engineers Journal,* vol. 123, no. 3, pp. 66-77, 2011.

[345] J. B. Famme, C. Gallagher and T. Raitch, "Performance-Based Design for Fleet Affordability," *Naval Engineers Journal,* vol. 121, no. 4, pp. 117-132, 2009.

[346] B. Langland, R. Leonard, R. Smart and R. A. Dougal, "Modeling and data exchange in a concurrent and collaborative design environment for electric ships," in *IEEE Electric Ship Technologies Symposium (ESTS)*, Alexandria, 2015.

[347] I. Mahmood, "A Verification Framework for Component Based Modeling and Simulation," KTH Royal Institute of Technology, Stockholm, 2013.

[348] K. B. Marais, J. Rivas, I. J. Tetzloff and W. A. Crossley, "Modeling the Impact of Maintenance on Naval Fleet Total Ownership Cost," in *IEEE International Systems Conference*, Orlando, 2013.

[349] I. C. Stults, "A Multi-Fidelity ANalysis Selection Method Using a COnstrained DIscrete Optimization Formulation," Georgia Institute of Technology, Atlanta, 2009.

[350] M. G. Fernandez-Godino, C. Park, N.-H. Kim and R. T. Haftka, "Review of multi-fidelity models," 12 June 2017. [Online]. Available: https://arxiv.org/pdf/1609.07196.pdf. [Accessed 31 July 2019].

[351] M. G. Fernandez-Godino, C. Park, N. H. Kim and R. T. Haftka, "Issues in Deciding Whether to Use Multi-fidelity Surrogates," *AIAA Journal*, vol. 57, no. 5, 2019.

[352] P. M. Gresho and R. L. Lee, "Don't Suppress the Wiggles - They're Telling you Something!," *Computers and Fluids*, vol. 9, pp. 223-253, 1981.

[353] PGPF, "U.S. Defense Spending Compared to Other Countries," Peter G. Peterson Foundation, 3 May 2019. [Online]. Available: https://www.pgpf.org/chart-archive/0053_defense-comparison. [Accessed 1 August 2019].

[354] A. Mehta, "Here's How Much Global Military Spending Rose in 2018," Defense News, 28 April 2019. [Online]. Available: https://www.defensenews.com/global/2019/04/28/heres-how-much-global-military-spending-rose-in-2018/. [Accessed 1 August 2019].

[355] K. Mizokami, "China Now Has More Warships Than the U.S.: But sometimes quantity doesn't trump quality.," Popular Mechanics, 1 August 2019. [Online]. Available: https://www.popularmechanics.com/military/navy-ships/a27532437/china-now-has-more-warships-than-the-us/. [Accessed 20 May 2019].

[356] R. O'Rourke, "Navy Force Structure and Shipbuilding Plans: Background and Issues for Congress," Congressional Research Service, Washington D.C., 2018.

[357] L. J. Peters, "The Joint Capabilities Integration and Development System: Its Impact on Air Force Acquisition Thirteen Years Later," Air University, Maxwell AFB, 2016.

[358] The MITRE Corporation, "Navy Future Fleet Platform Architecture Study," McLean, 2016.

[359] A. Bresler, "Bridging the Gap: Improving DoD-Backed Innovation Programs to Enhance the Adoption of Innovative Technology Throughout the Armed Services," in *Proceedings of the Fifteenth Annual Acquisition Research Symposium* - *Volume I*, Monterey, 2018.

[360] M. Mackin, D. Moldafsky, G. Campbell, L. Fish, L. Greifner, K. Hassinger, C. J. Madar, K. Patton, J. Rastler, A. Steele, R. Sun and H. Tran, "Littoral Combat Ship: Knowledge of Survivability and Lethality Capabilities Needed Prior to Making Major Funding Decisions," United States Government Accountability Office, Washington D.C., 2015.

[361] J. Dreo, A. Petrowski, P. Siarry and E. Taillard, "Collective behavior of social insects," in *Metaheuristics for Hard Optimization: Methods and Case Studies*, Berlin, Springer-Verlag, 2006, pp. 124-128.

[362] P. T. Biltgen, "A Methodology for Capability-Based Technology Evaluation for Systems-of-Systems," Georgia Institute of Technology, Atlanta, 2007.

[363] R. Burkhart, "MBSE Wiki," Object Management Group, Inc., 28 January 2019. [Online]. Available: omgwiki.org/MBSE/doku.php?id=start. [Accessed 23 August 2019].

[364] INCOSE, IEEE Computer Society, Stevens Institute of Technology, "Model-Based Systems Engineering," Systems Engineering Body of Knowledge, 1 June 2019. [Online]. Available: sebokwiki.org/wiki/Model-Based_Systems_Engineering_(MBSE)_(glossary). [Accessed 26 August 2019].

[365] C. Szabo and L. Birdsey, "Validating Emergent Behavior in Complex Systems," in *Advances in Modeling and Simulation: Simulation Foundations, Methods and Applications,* A. Tolk, J. Fowler, G. Shao and E. Yücesan , Eds., Springer, Cham, 2017, pp. 47-62.

[366] R. K. Sawyer, "Durkheim's Dilemma: Toward a Sociology of Emergence," *Sociology Theory,* vol. 20, no. 2, pp. 227-247, 2002.

[367] S. A. Sheard, "Systems Engineering Complexity in Context," *INCOSE International Symposium,* vol. 23, no. 1, pp. 1145-1158, 2013.

[368] R. Sapolsky, *Chaos and Reductionism*, Stanford, CA: YouTube, 2010.

[369] R. Sapolsky, *The Biology of Our Best and Worst Selves*, Vancouver: TED, 2017.

[370] Basic Energy Sciences Advisory Committee, "From Quanta to the Continuum: Opportunities for Mesoscale Science," Argonne National Laboratory - United States Department of Energy, Lemont, 2012.

[371] R. I. Damper, "Editorial for the Special Issue on 'Emergent Properties of Complex Systems': Emergence and levels of abstraction," *International Journal of Systems Science*, vol. 31, no. 7, pp. 811-818, 2000.

[372] C.-C. Chen, C. D. Clack and S. B. Nagl, "Identifying Multi-Level Emergent Behaviors in Agent-Directed Simulations using Complex Event Type Specifications," *Simulation*, vol. 86, no. 1, pp. 41-51, 2010.

[373] V. R. Basili, "Software Modeling and Measurement: The Goal/Question/Metric Paradigm," [Online]. Available: https://www.cs.umd.edu/~basili/publications/technical/T78.pdf. [Accessed 17 October 2019].

[374] Chief Information Officer, "DoDAF Meta-Model (DM2)," Department of Defense, August 2010. [Online]. Available:

https://dodcio.defense.gov/Library/DoD-Architecture-Framework/dodaf20_dm2/. [Accessed 20 August 2019].

[375] U. Wilensky, "NetLogo," Center for Connected Learning and Computer-Based Modeling - Northwestern University, 1999. [Online]. Available: https://ccl.northwestern.edu/netlogo/. [Accessed 20 October 2019].

[376] D. Miner and N. Kasch, "SwarmVis: a Tool for Visualizing Swarm Systems," 2008. [Online]. Available: https://pdfs.semanticscholar.org/d506/ad8a491953f965d88e5d864e28974f1f402f.pdf?_ga=2.163089212.1443278090.1571635047-1782275441.1571352389. [Accessed 21 October 2019].

[377] G. Minati and I. Licata, "Meta-structural properties in collective behaviors," *International Journal of General Systems,* vol. 41, no. 3, pp. 289-311, 2012.

[378] P. Bak, C. Tang and K. Weisenfeld, "Self-Organized Criticality," *Physical Review A,* vol. 38, no. 1, pp. 364-374, 1988.

[379] I. Kesternich, B. Siflinger, J. P. Smith and J. K. Winter, "The Effects of World War II on Economic and Health Outcomes Across Europe," *The Review of Economics and Statistics,* vol. 96, no. 1, pp. 103-118, 2014.

[380] "Computational Complexity of Mathematical Operations," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Computational_complexity_of_mathematical_operations. [Accessed 24 June 2019].

[381] S. B. Pope, "Wall Flows," in *Turbulent Flows*, New York, Cambridge University Press, 2000, pp. 264-332.

[382] K. Matouš, M. G. Geers, V. G. Kouznetsova and A. Gillman, "A review of predictive nonlinear theories for multiscale modeling of heterogeneous materials," *Journal of Computational Physics,* vol. 330, no. 1, pp. 192-220, 2017.

[383] R. Abbott, "Emergence Explained: Abstractions," *Complexity,* vol. 12, no. 1, pp. 13-26, 2006.

[384] T. M. Larsen, "John Boyd and the AirLand Battle Doctrine," United States Army Command and General Staff College, Fort Leavenworth, 2012.

[385] W. M. Spears, "NetLogo and Physics," in *Physicomimetrics: Physics-Based Swarm Intelligence*, W. M. Spears and D. F. Spears, Eds., Berlin, Springer, 2012, pp. 27-53.

[386] E. Levine, Writer, *How I Made Friends with Reality.* [Performance]. TED, 2018.

[387] E. Shafak, Writer, *The Politics of Fiction.* [Performance]. TEDGlobal, 2010.

[388] P. Rendell, "Turing Universality in the game of life," in *Collision-Based Computing*, A. Adamatzky, Ed., New York, Springer, 2002.

[389] J. A. Olsen, "Boyd Revisited: A Great Mind with a Touch of Madness," *Air Power History*, vol. Winter, pp. 6-16, 2016.

[390] F. C. Spinney, "Introduction to the Strategic Theories of John Boyd," Blaster (blog) - Blogspot, 29 April 2015. [Online]. Available: chuckspinney.blogspot.com/p/compendium-colonel-john-boyds.html. [Accessed 22 November 2020].

[391] Marine Corps University Library, "The Personal Papers of Colonel John R. Boyd (USAF) - Part II," Marine Corps University, Quantico, 2018.

[392] A. Tolish, "General Relativity and the Newtonian Limit," VIGRE, Department of Mathematics, University of Chicago, 2010. [Online]. Available: https://math.uchicago.edu/~may/VIGRE/VIGRE2020/REUPapers/Tolish.pdf. [Accessed 1 December 2020].

[393] S. Carroll, "Lecture Notes on General Relativity," California Institute of Technology, 17 August 2001. [Online]. Available:

https://ned.ipac.caltech.edu/level5/March01/Carroll3/Carroll4.html. [Accessed 1 December 2020].

[394] R. Cilibrasi, "Statistical Inference Through Data Compression," Institute for Logic, Language, and Computation - Universiteit van Amsterdam, Amsterdam, 2007.

[395] M. Rayman, Interviewee, *How Many Decimals of Pi Do We Really Need?.* [Interview]. 16 March 2016.

[396] R. D. Rupert, "Functionalism, Mental Causation, and the Problem of Metaphysically Necessary Effects," *Noûs,* vol. 40, no. 2, pp. 256-283, 2006.

[397] I. C. Baianu, R. Brown and J. F. Glazebrook, "Categorical Ontology of Complex Spacetime Structures: The Emergence of Life and Human Consciousness," *Axiomathes,* vol. 17, pp. 223-352, 2007.

[398] J. J. Macionis and L. M. Gerber, Sociology, 7th ed., Toronto: Pearson Prentice Hall, 2010.

[399] W. A. Strauss, Partial Differential Equations, 2nd ed., Hoboken: John Wiley & Sons, Inc., 2008.

[400] ritvikmath, "Time Series Talk: Autoregressive Model," YouTube, 11 April 2019. [Online]. Available: https://youtu.be/5-2C4eO4cPQ.

[401] M. X. Cohen, "Granger causality (prediction)," YouTube, 21 December 2019. [Online]. Available: https://youtu.be/XqsSB_vpHLs.

[402] B. Lambert, "Autoregressive Order one process introduction and example," YouTube, 16 September 2013. [Online]. Available: https://youtu.be/AN0a58F6cxA.

[403] J. Pearl, "The Mathematics of Causal Relations," Oxford University Press, Oxford, 2010.

[404] G. Coring-Hassan, "Inferring Causality in Agent-Based Simulations - Literature Review," 18 December 2018. [Online]. Available: https://arxiv.org/abs/1901.04836.

[405] MathWorks, "Econometrics Toolbox," MathWorks, 2020. [Online]. Available: https://mathworks.com/help/econ. [Accessed September 2020].

[406] E. R. Dougherty, "Science without Validation in a World without Meaning," *American Affairs*, vol. IV, no. 2, 2020.

[407] J. D. Halley and D. A. Winkler, "Consistent Concepts of Self-organization and Self-assembly," *Complexity,* vol. 14, no. 2, pp. 10-17, 2008.

[408] J. Lin, E. Keogh, S. Lonardi and P. Patel, "Finding Motifs in Time Series," in *2002*, Alberta, SIGKDD '02.

[409] J. P. Müller, D. Phan and F. Varenne, "An ontological perspective on the action - structure debate for agent-based framework: the case of "following a rule"," January 2015. [Online]. Available: https://www.researchgate.net/publication/264869676_An_ontological_perspective_on_the_action_-_structure_debate_for_agent-based_framework_the_case_of_following_a_rule. [Accessed 8 September 2019].

[410] B. Umut, "Causal Emergence and Epiphenomenal Emergence," *Erkenntnis,* vol. 85, pp. 891-904, 2020.

[411] C. Emmeche, S. Køppe and F. Stjernfelt, "Explaining Emergence: Towards an Ontology of Levels," *Journal for General Philosophy of Science,* vol. 28, no. 1, pp. 83-119, 1997.

[412] J. Lawhead, "Self-Organization, Emergence, and Constraint in Complex Natural Systems," 5 February 2015. [Online]. Available: https://arxiv.org/abs/1502.01476.

[413] C. T. Kohl, "Buddhism and Quantum Physics," *Contemporary Buddhism,* vol. 8, no. 1, pp. 69-82, 2007.

[414] E. Nozari, Y. Zhao and J. Cortés, "Network Identification With Latent Nodes via Autoregressive Models," *IEEE Transactions on Control of Network Systems,* vol. 5, no. 2, pp. 722-736, 2018.

[415] J.-L. Voirin, S. Bonnet, D. Exertier and V. Normand, "Simplifying (and enriching) SysML to perform functional analysis and model instances," in *26th Annual INCOSE International Symposium*, Edinburgh, 2016.

[416] B. Berklich, "Functional Decomposition Process," 2 June 2011. [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a543217.pdf.

[417] C. Hooker, "Introduction to Philosophy of Complex Systems Part A: Towards a Framework for Complex Systems," in *Handbook of the Philosophy of Science: Philosophy of Complex Systems*, 1st ed., vol. 10, C. Hooker, D. M. Gabbay, P. Thagard and J. Woods, Eds., Waltham, Elsevier BV, 2011, pp. 3-90.

[418] M. O'Dowd, "What Happens During a Quantum Jump?," PBS Space Time, 12 January 2021. [Online]. Available: https://youtu.be/j5HyMNNSGqQ. [Accessed 12 January 2021].

[419] P. W. Melançon, "Decomposition by Experts," Naval Postgraduate School, Monterey, 2008.

[420] J. P. Mo, C. Bil and A. Sinha, Eds., "Systems design," in *Engineering Systems Acquisition and Support*, Cambridge, Woodhead Publishing / Elsevier Ltd., 2015, pp. 51-73.